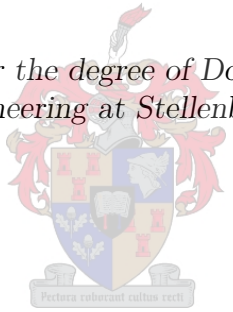
Automatic Flight Envelope Recovery for Large Transport Aircraft

by

Jacobus Adriaan Albertus Engelbrecht

Dissertation presented for the degree of Doctor of Philosophy in the Faculty of Engineering at Stellenbosch University



Department of Electrical and Electronic Engineering
University of Stellenbosch
Private Bag X1, 7602 Matieland, South Africa

Promoter: Dr Iain Peddle, Prof Thomas Jones

December 2016

Declaration

By submitting this dissertation electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the owner of the copyright thereof (unless to the extent explicitly otherwise stated) and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

December 2016

Date: .....

Abstract

This thesis presents the design of an automatic flight envelope recovery system for large transport aircraft. The flight envelope of the aircraft is conceptualised to consist of the aerodynamic envelope, the attitude envelope, the flight vector envelope, and the structural integrity envelope. The time scale separation between the aircraft's fast rigid body rotational dynamics and its slower point mass translational dynamics is exploited to separate the flight envelope recovery into an inner-loop recovery of the angular rates and aerodynamic envelope, and an outer-loop recovery of the attitude, flight path angle, and airspeed while remaining within the structural integrity envelope.

Two approaches to flight envelope recovery are proposed. The first approach uses a passive method to recover the aerodynamic envelope using the natural stability of the aircraft and then uses the conventional flight control laws, with their flight envelope protection functions, to recover the aircraft attitude and flight vector. The second approach uses an active method to recover the aerodynamic envelope using a Lyapunov-based inner-loop controller and uses an outer-loop controller based on optimal control theory to recover the aircraft attitude, flight path angle and airspeed while minimising the altitude loss.

The automatic flight envelope recovery is verified in simulation on the NASA Generic Transport Model (GTM), a wide-envelope aircraft model that is able to model the flight mechanics of large transport aircraft in out-of-envelope conditions.

Opsomming

Hierdie tesis beskryf die ontwerp van 'n outomatiese vlug-omhullende herstel stelsel vir groot transport vliegtuie. Die vlug-omhullende van die vliegtuig word konseptualiseer om te bestaan uit die aerodinamiese-omhullende, die oriëntasie-omhullende, die vlugvektor-omhullende, en die strukturele-integriteit-omhullende. Die tydskaalskeiding tussen die vliegtuig se vinnige starreliggaam-rotasiedinamika en sy stadiger puntmassa-translasiedinamika word uitgebuit om die vlug-omhullende herstel te skei in 'n binnelus herstel van die hoektempos en aerodinamiese-omhullende, en 'n buitelus herstel van die oriëntasie, vlugpadhoek, en lugspoed terwyl daar binne die strukturele-integriteit-omhullende gebly word.

Twee benaderings tot vlug-omhullende herstel word voorgestel. Die eerste benadering gebruik 'n passiewe metode om die aerodinamiese-omhullende te herstel deur gebruik te maak van die natuurlike stabiliteit van die vliegtuig, en gebruik daarna die konvensionele vlugbeheerwette, met hulle beskermingsfunksies, om die vliegtuig oriëntasie, vlugpadhoek en lugspoed te herstel. Die tweede benadering gebruik 'n aktiewe metode om die aerodinamiese omhullende te herstel deur gebruik te maak van 'n Lyapunov-gebaseerde binnelusbeheerder en gebruik 'n buitelus beheerder gebaseer op optimale beheerteorie om die vliegtuig oriëntasie, vlugpadhoek en lugspoed te herstel terwyl dit die hoogteverlies minimeer.

Die outomatiese vlug-omhullende herstel word geverifeer in simulasie op die NASA Generic Transport Model (GTM), 'n wye-omhullende vliegtuigmodel wat in staat is om die vlugmeganika van groot transport vliegtuie in toestande buite die normale vlug-omhullende te modelleer.

Acknowledgements

I would like to thank the following people and organisations for their support and contributions towards this dissertation:

- My supervisors, Dr Iain Peddle and Prof Thomas Jones, who suggested that I pursue this topic, who asked the difficult questions and who pointed me in the direction of possible answers. Special thanks to Prof Thomas Jones for reading through my thesis and providing valuable feedback.
- My amazing wife, Madri, for her love and support, both emotionally and practically, to enable me to complete this PhD thesis. Thank you especially for creating opportunities for me to go on week-long PhD retreats to spend continuous, focussed time on my research and thesis writing.
- My two daughters, Amé and Mira, who are the apples of my eye and who bring so much happiness into my life.
- My parents, Japie and Suzanne Engelbrecht, for raising me and believing in me. My father instilled a sense of wonder for "how things work" in me, and my mother taught me how to study and how to keep on studying until I knew my work.
- My brothers, Herman and Schalk, who are the best brothers that anyone can have, and who both completed their PhDs long before me.
- All my family, friends and colleagues who kept on asking when I was going to submit my PhD. Special mention goes to Herman Engelbrecht, Thinus Booysen, and Johan Booysen, who were particulary persistent. Special thanks also to Thomas Jones, Iain Peddle and Petrie Meyer for giving me the final push.
- Our friends, Gerhard en Aneleh Fourie-Le Roux, and Gideon and Mariheca Strydom-Otto, for their friendship over the years, and for providing welcome and needed distractions.
- Dr Etienne Coetzee at Airbus Filton, for making us aware of the NASA Generic Transport Model, and for assisting with the bifurcation and continuation software.
- Dr Stephen Gill and Prof Mark Lowenberg from Bristol University for our informal collaboration on the bifurcation analysis of the NASA Generic Transport Model.
- Simon Pauck, who modified the NASA GTM Simulink model to be compatible with the Matlab Dynamical Systems toolbox and generated the bifurcation diagrams that are used to support the passive approach to aerodynamic envelope recovery.
- The technical experts at Airbus Toulouse, Guilhem Puyou, Emmanuel Cortet, Pierre Fabre, and Josep Boada-Bauxell, for their technical guidance and feedback on the flight envelope recovery research.

- The NASA Langley Research Center for making the GTM simulation model available for this research.
- Prof Jan du Plessis, for his postgraduate lectures on nonlinear control systems that have stuck with me over the years, and from whom I learned to appreciate the physical interpretation of the mathematics of feedback control.
- My mentor and colleague, Prof Herman Steyn, from whom I learned almost everything I know about satellite attitude determination and control systems, and who taught me to be critical about my work.
- My colleagues, Corné van Daalen, Johann Treurnicht, Iain Peddle, Thomas Jones, and Herman Steyn who deepened my understanding of the theory and practical application of feedback control systems.
- All the postgraduate students that I have supervised over the past eight years who have explored different aspects of flight control with me through their Master's degree topics.
- My "skripsie" student, Jannes Engelbrecht (no relation), who explored optimal attitude and flight vector recovery using dynamic programming with me through his final year project, and who walked into the "90 degree bank angle problem".
- My fellow PhD students, Steven Kriel and Willem Jordaan, who were in the trenches with me.
- Chris Jaquet and Corné van Daalen, who encouraged me to use LateX to write my thesis. The clean interface has allowed me to focus on the content without worrying about the formatting, and all the little LateX "puzzles" to be solved while writing the thesis have been a welcome distraction.
- Evert Trollip, who allowed me to borrow his scalable vector graphics images of a commercial airliner and use them in my own aircraft diagrams.
- Newton, Euler, Lyapunov, La Salle, and Bellman, for creating their wonderful mathematical tools for me to play with and apply to the problem of flight envelope recovery.
- The little coastal town of Struisbaai, where I could get away from everything and focus on my PhD research and thesis writing. If it were not for several week-long PhD retreats to Struisbaai over the past two years, I am sure that I would never have completed my PhD thesis for submission.

Contents

Nomenclature

Acronyms

ADMIRE	Aero-Data Model in a Research Environment
AirSTAR	Airborne Subscale Transport Aircraft Research
GAM	Generic Aerodata Model
GTM	Generic Transport Model
HARV	High Angle-of-attack Research Vehicle
HHIRM	Hypothetical High Incidence Research Model
NASA	National Aeronautics and Space Administration
PAARS	Pilot Activated Automatic Recovery System
QLC	Quantitative Loss-of-Control Criteria
SUPRA	Simulation of Upset Recovery in Aviation
UAV	Unmanned Aerial Vehicle

Symbol Conventions

x	Scalar
$\mathbf{x}$	Vector
$\mathbf{X}$	Matrix
$\mathcal{X}$	Set
$\dot{x}$	Derivative of x

Constants

g	Gravitational acceleration

Aircraft Parameters

m	Mass
$\boldsymbol{I_B}$	Moment of inertia matrix of the aircraft body
S, $\bar{c}$, b	Wing surface area, mean aerodynamic chord, and wing span

Aircraft Dynamics

X, Y, Z	Coordinates of force vector in body axes (axial, lateral, and normal force)
L, M, N	Coordinates of moment vector in body axes (roll, pitch, and yaw moment)
U, V, W	Coordinates of linear velocity vector in body axes (axial, lateral, and normal velocity)
P, Q, R	Coordinates of angular velocity vector in body axes (roll, pitch, and yaw rates)

δ_A, δ_R, δ_R	Aileron, rudder, and elevator deflections. A positive deflection is defined as one that produces a negative moment.
$\overline{V}$, α, β	Airspeed magnitude, angle of attack, and sideslip angle
N, E, D	Coordinates of position vector in inertial axes (north, east and down position)
Φ, Θ, Ψ	Euler 3-2-1 attitude parameters of body axis system relative to inertial axis system (roll, pitch, and yaw angle)

Wide-Envelope Aerodynamic Model

$\overline{\mathbf{\Omega}}$	Total angular rate vector
$\overline{\mathbf{\Omega}}_{xz}$	Projection of total angular rate vector into body-axis xz-plane
p_b, q_b, r_b	Body-axis roll rate, pitch rate and yaw rate
ω_{ss}	Steady-state angular rate component assigned to wind-axis roll rate
p_{osc}, q_{osc}, r_{osc}	Oscillatory angular rate components assigned to body-axis roll rate, body-axis pitch rate, and body-axis yaw rate
$\hat{\omega}_{ss}$	Non-dimensional steady-state wind-axis roll rate component
$\hat{p}_{osc}$, $\hat{q}_{osc}$, $\hat{r}_{osc}$	Non-dimensional oscillatory roll rate component, pitch rate component, and yaw rate component
C_X, C_Y, C_Z	Aerodynamic force coefficients along body x-axis, y-axis, and z-axis
C_l, C_m, C_n	Aerodynamic moment coefficients about body x-axis, y-axis, and z-axis (pitching moment, rolling moment, and yawing moment)
C_L, C_D	Aerodynamic lift coefficient and drag coefficient in wind axes

Bifurcation Theory

$\mathbf{f}$	Vector of ordinary nonlinear differential equations in vector form
$\mathbf{x}$	State vector
$\mathbf{u}$	Control input vector
$\mathbf{p}$	Vector of system parameters
$\mathbf{x}^*$	Equilibrium state
λ	Continuation parameter
$J(\mathbf{x}^*)$	Jacobian matrix at equilibrium state

Conventional Control Laws

$\overline{V}_{\text{ref}}$	Airspeed reference
$n_{L,\text{ref}}$	Normal load factor reference
γ_{ref}	Flight path angle reference
p_{ref}	Roll rate reference
ϕ_{ref}	Bank angle reference

Lyapunov Stability Theory

$\mathbf{x}$	State vector
$V(\mathbf{x})$	Energy-like function
$\dot{V}(\mathbf{x})$	Time rate of change of energy-like function
$\mathbf{\Omega}$	Domain of attraction of an equilibrium state

$\mathbf{u}$	Input vector
$\mathbf{g}$	Control law that makes the energy-like function $V(\mathbf{x})$ a Lyapunov function of the system

Lyapunov-Based Controller

α_{REF}	Angle of attack reference
β_{REF}	Sideslip angle reference
$P_{W_{REF}}$	Wind-axis roll rate reference
$(P, Q, R)_{REF}$	Angular rate references in body axes
Q_W, R_W	Wind-axis pitch rate and yaw rate
a_{Wz}, a_{Wy}	Total normal and lateral acceleration of the aircraft in wind axes
$\overline{V}_{\text{ground}}$	Ground speed
L, M, N	Rolling, pitching, and yawing moments coordinated in body axes (if the subscripts A, E, or G are added, they denote the aerodynamic moments, engine moments, and gravitational moments, respectively
$C_l()$, $C_m()$, $C_n()$	Non-dimensional aerodynamic coefficient functions for rolling, pitching and yawing moments
P_B^{WI}, Q_B^{WI}, R_B^{WI}	Angular rate components of the wind axis system relative to the inertial axis system, coordinated in the body axis system
K	Lyapunov control law gain that determines the rate of convergence of the "energy" of the system
k_α, k_β	Lyapunov control law gains that determine the relative weighting of the angle of attack and the sideslip angle within the state vector
k_P, k_Q, k_R	Lyapunov control law gains that determine the relative weighting of the body angular rate within the state vector

Point Mass Translational Dynamics

V_N, V_E, V_D	Cartesian coordinates of aircraft velocity vector in inertial axes (north, east and down velocity)
$\overline{V}$, γ, ψ_W	Airspeed magnitude, flight path angle, flight path heading
ϕ_W	Wind-axis bank angle
h	Altitude
L, D, Y	Coordinates of force vector in wind axes (lift, drag, and side force)
T	Thrust force
P_W	Wind-axis roll rate
ρ	Air density
S	Wing surface area
C_L, C_Y, C_D	Non-dimensional aerodynamic coefficients for lift, side force, and drag
$\frac{1}{2}\rho\overline{V}^2$	Dynamic pressure
T, T_C, τ	Thrust, thrust command, and engine time constant

Optimal Control Theory

$\mathbf{x}$, $\mathbf{u}$	State vector and control input vector
$\mathbf{x}(t)$, $\mathbf{u}(t)$	Candidate state trajectory and candidate control input signal

$\mathbf{f}()$	Set of nonlinear, time-invariant differential equations of the system
$g()$	State transition cost function
$h()$	Terminal state cost function
$\mathcal{X}, \mathcal{U}$	Sets of admissible states and admissible control inputs
$\mathcal{X}_\mathrm{f}$	Set of admissible final states
$\mathbf{x}^*(t)$, $\mathbf{u}^*(t)$	Optimal state trajectory and optimal control input signal
$\mathbf{x}(k)$	Current state at sampling instant k
$\mathbf{x}(k+1)$	Next state at sampling instant $k+1$
$\mathbf{x}^*(k+1)$	Optimal next state at sampling instant $k+1$
$\mathbf{u}(k)$	Control input at sampling instant k
$\mathbf{u}^*(k)$	Optimal control input at sampling instant k

Optimal Control Formulation

Δh	Altitude change during recovery
$\overline{V}_{\min}$, $\overline{V}_{\max}$	Minimum and maximum admissible airspeeds
$\gamma_{\min}$, $\gamma_{\max}$	Minimum and maximum admissible flight path angles
$-\phi_{W\max}$, $\phi_{W\max}$	Minimum and maximum admissible wind-axis bank angles
$\alpha_{\min}$, $\alpha_{\max}$	Minimum and maximum admissible angle of attack inputs
$T_{\min}$, $T_{\max}$	Minimum and maximum admissible thrust inputs
$-P_{W\max}$, $P_{W\max}$	Minimum and maximum admissible wind-axis roll rate inputs
$n_{L\min}$, $n_{L\max}$	Minimum and maximum admissible normal load factor
$\overline{V}_{f_{\min}}$, $\overline{V}_{f_{\max}}$	Minimum and maximum admissible final airspeeds
J	Cost function to be optimised
J^h	Cost function representing the total altitude lost during the recovery
$J^{\overline{V}}$	Cost function representing the maximum velocity reached during the recovery
J^ϕ	Cost function representing the time integral of the bank angle
t_0	Initial time instant
t_f	Final time instant

Dynamic Programming Theory

$\mathbf{X}_q$	Array of quantised state vector values
$\mathbf{U}_q$	Array of quantised input vector values
Δt	Size of the discrete time step.
n	Number of quantised states
N	Number of discrete time steps
k	Index of current time step
i	Index of quantised step at current time index k
j	Index of quantised step at next time index k
J_{ij}	Total path cost from current state $\mathbf{x}_i(k)$ via next state $\mathbf{x}_j(k+1)$
ΔJ_{ij}	Incremental path cost of transitioning from current state $\mathbf{x}_i(k)$ to next state $\mathbf{x}_j(k+1)$
J_j	Total path cost from next state $\mathbf{x}_j(k+1)$ to a final state

$\mathbf{u}_{ij}$	Control input that transitions the system from state $\mathbf{x}_i$ at time k to state $\mathbf{x}_j$ at time $k+1$
$\mathbf{J}^*_{n\times N}$	Data structure for storing the optimal cost $J^*_{i,k}$ from each state $\mathbf{x}_i$ at time instant k
$\mathbf{j}^*_{n\times N}$	Data structure for storing the optimal next state $j^*_{i,k}$ from each state $\mathbf{x}_i$ at time instant k
$\mathbf{U}^*_{n\times N}$	Data structure for storing the optimal control input $\mathbf{u}^*_{i,k}$ from each state $\mathbf{x}_i$ at time instant k

Dynamic Programming Solution

$\mathbf{X}_q$	Array of quantised state vector values
$\overline{\mathbf{V}}_q$	Array of quantised airspeed values
$\mathbf{\Gamma}_q$	Array of quantised flight path angle values
$\mathbf{\Phi_W}_q$	Array of quantised wind-axis bank angle values
$\mathbf{J}^{h*}$	Optimal altitude cost array
$\mathbf{J}^{\overline{V}*}$	Optimal airspeed cost array
$\mathbf{J}^{\phi*}$	Optimal bank angle cost array
$\mathbf{j}^*$	Optimal next state index array
$\mathbf{U}^*$	Optimal control input array

Integrated Flight Envelope Recovery

$\overline{V}_{\text{crnt}}$	Current airspeed state
γ_{crnt}	Current flight path state
$\Phi_{W\,\text{crnt}}$	Current wind-axis bank angle state
α_{cmd}	Angle of attack command
$P_{W\,\text{cmd}}$	Wind-axis roll rate command
T_{cmd}	Thrust command
γ^*_{next}	Optimal next flight path angle state
$\overline{V}^*_{\text{next}}$	Optimal next airspeed state
γ^*_{next}	Optimal next flight path angle state
α^*_{cmd}	Optimal angle of attack command
$P_{W}{}^*_{\text{cmd}}$	Optimal wind-axis roll rate command
T^*_{cmd}	Optimal thrust command
α_{ref}	Angle of attack reference for Lyapunov controller
$P_{W\,\text{ref}}$	Wind-axis roll rate reference for Lyapunov controller
β_{ref}	Sideslip angle reference for Lyapunov controller
γ_{ref}	Flight path angle reference for conventional control law
$n_{L\,\text{cmd,fb}}$	Total normal load factor command
$n_{L\,\text{cmd,fb}}$	Normal load factor command, feedback component
$n_{L\,\text{cmd,ffwd}}$	Normal load factor command, feedforward component
Φ_{ref}	Bank angle reference for conventional control law
$P_{\text{cmd,fb}}$	Total roll rate command
$P_{\text{cmd,fb}}$	Roll rate command, feedback component
$P_{\text{cmd,ffwd}}$	Roll rate command, feedforward component

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

The flight envelope of an aircraft is the domain of flight conditions in which the aircraft can safely be operated without exceeding its aerodynamic and structural limits [4]. The flight envelope is often defined in terms of the limiting boundaries of altitude, airspeed, angle of attack, and normal load factor.

Flight envelope protection is an extension of an aircraft's flight control system that prevents the pilot from giving control commands that would force the aircraft to exceed its flight envelope. For example, if the pilot pulls back on the control stick to pitch the aircraft nose up, the flight envelope protection will prevent the aircraft nose from pitching up beyond the stalling angle of attack. Even if the pilot tried to apply more and more rearward control stick, the flight envelope protection would cause the aircraft to ignore this command. The flight envelope protection therefore allows the pilot to give maximum control commands in emergency situations while protecting him from inadvertently causing the aircraft to exit its safe operating limits.

Despite the fact that envelope protection functions are implemented on modern commercial aircraft, situations still arise where the aircraft exits the flight envelope due to environmental factors, pilot error, or component failures. Fortunately, these occurences are quite rare. Unfortunately, when they do occur, the flight conditions have already exceeded the valid operating domain of the flight control system, and the full responsibility for controlling the aircraft falls on the pilot. The envelope protection functions are only designed to prevent the aircraft from exiting the flight envelope, and not for recovering the flight envelope after it has already exited.

A need therefore exists for a flight envelope recovery function that assists the pilot in recovering the aircraft to its flight envelope after a severe upset. The development of such a flight envelope recovery function would represent a valuable contribution to commercial aviation safety, since its implementation would potentially reduce the number of accidents caused by aircraft flight envelope upset.

1.2 Research Objectives

The objective of this research is to propose a flight envelope recovery strategy for large transport aircraft and to design and evaluate novel guidance and flight control laws to recover the aircraft state to the normal flight envelope from out-of-envelope conditions. The flight envelope recovery flight control system will assume that the aircraft has fully functional control surfaces and sensors, but that it finds itself in flight envelope upset conditions due to environmental factors, pilot error, or transient component failures. The purpose of this research is not to address control surface failures or sensor failures, for which a large body of research on fault-tolerant control and fault detection and isolation already exists.

1.3 Primary Contributions

The research presented in this thesis makes the following primary contributions:

- Automatic flight envelope recovery strategies for large transport aircraft are developed. The majority of the previous research on flight envelope recovery is applied to high performance fighter aircraft, and very little research has been performed for large transport aircraft. Fighter aircraft are designed to operate over much wider flight envelopes than large transport aircraft, and are also equipped with specialised actuators that have much greater control authority. Previous research also tends to focus more on flight envelope *protection* and less on flight envelope *recovery*.

- The flight envelope recovery is conceptualised to encompass high angular rate recovery, aerodynamic envelope recovery, attitude recovery, flight path angle recovery, and airspeed recovery while respecting the load factor and maximum airspeed constraints of the structural integrity envelope.

- The time scale separation between the aircraft's fast rigid body rotational dynamics and it's slow point mass translational dynamics is exploited to conceptually split the flight envelope recovery into an "inner loop recovery" of the angular rates and aerodynamic envelope, and an "outer loop recovery" of the attitude, flight path angle, and airspeed.

- A passive aerodynamic envelope recovery method is proposed that uses the natural angular rate damping of the aircraft to recover from high angular rates, and uses the natural stability of the aircraft to recover the angle of attack and sideslip angle. The passive approach does not require explicit knowledge of the forces and moments produced by the control surfaces, and does not rely on the availability of wide-envelope anemometric sensor measurements. A bifurcation analysis is performed to identify the desirable stable equilibria of the aircraft inside the aerodynamic envelope, and to check for undesirable stable equilibria outside the aerodynamic envelope. The regions of attraction of the stable equilibria, both inside and outside the aerodynamic envelope, are explored with nonlinear simulations by generating state trajectories and by performing Monte Carlo simulations.

- A state machine based approach for attitude, flight path angle and airspeed recovery is proposed that recovers the flight envelope in stages and uses the passive aerodynamic envelope recovery as the first stage of the recovery. Once the aerodynamic envelope is recovered, the control surfaces produce forces and moments that behave linearly again, and the normal flight control laws and protection laws are available to perform the attitude, flight path angle and airspeed recovery.

- An active aerodynamic envelope recovery method is proposed that employs a Lyapunov-based inner-loop controller to actively recover the angular rates, the angle of attack, and the sideslip angle from out-of-envelope conditions using the control surfaces of the aircraft. A novel Lyapunov function is proposed that represents the state of the fast rotational dynamics of the aircraft as an energy-like function. Lyapunov-based control laws are then derived to control the time derivative of the energy to always be negative and proportional to the energy, thus providing exponential Lyapunov stability. The resulting Lyapunov-based inner-loop controller not only actively recovers the angular rates and the aerodynamic envelope, but allows the angle of attack, sideslip angle, and wind-axis roll rate to be actively controlled in an extended aerodynamic envelope.

- An optimal attitude and flight vector recovery guidance law is proposed that simultaneously recovers the bank angle, flight path angle, and airspeed of the aircraft from upset conditions while minimising the total altitude lost during the recovery maneuver. The

recovery problem is formulated as an optimal control problem and then solved using a dynamic programming algorithm to find the optimal state trajectories and the optimal sequence of control inputs. The aircraft dynamics is simplified to the reduced-order model describing the slower point mass translational dynamics of the aircraft, while the fast rotational dynamics is abstracted through time scale separation. The reduced-order model of the aircraft dynamics makes the optimal control problem tractable to be solved with dynamic programming.

- The inner loop Lyapunov control law and the outer loop optimal attitude and flight vector guidance law are combined into a single integrated flight envelope recovery system which is verified in simulation on the NASA Generic Transport Model.

1.4 Definition of Flight Envelope Upset

Flight envelope upset is associated with high angular rates, high angles of attack and sideslip angles, unusual pitch and bank angles, airspeeds that are either too low or too high, and high normal load factors [1]. For the purpose of this thesis, we conceptualise the flight envelope of the aircraft to consist of the following sub-envelopes:

1. The **Aerodynamic Envelope** (angle of attack vs. sideslip angle): The boundaries of the aerodynamic envelope are defined by the ranges of angle of attack and sideslip angle within which the aircraft was designed to operate. For angles of attack and sideslip angles that exceed the aerodynamic envelope, the aerodynamics becomes very nonlinear, and the aircraft response to control inputs differs from what the pilot expects.

2. The **Attitude Envelope** (bank angle vs. pitch angle): The boundaries of the attitude envelope are defined by the ranges of bank angle and pitch angle that are expected to be encountered in normal flight. The generally accepted industry definition is that bank angles exceeding $\pm 45°$, and pitch angles exceeding $10°$ nose down or $25°$ nose up are considered unusual attitudes. Note that is possible for the aircraft to violate the attitude envelope without violating the aerodynamic envelope.

3. The **Flight Vector Envelope** (flight path angle vs. airspeed): The boundaries of the flight vector envelope are defined by the ranges of flight path angle and airspeed that are expected to be encountered in normal flight. A large positive flight path angle implies that the aircraft is climbing steeply, which will eventually lead to an underspeed condition if their is insufficient thrust available to maintain the climb. A large negative flight path angle implies that the aircraft is descending steeply, which will eventually lead to an overspeed condition or impact with the ground. A low airspeed implies that the aircraft is close to the stall speed, which is the airspeed at which the aircraft cannot produce enough lift to maintain level flight. A high airspeed implies that the aircraft is close to its maximum operating speed, where excessive aircraft vibrations develop which put the aircraft structural integrity at risk.

4. The **Structural Integrity Envelope** (normal load factor vs. airspeed): The boundaries of the structural integrity envelope are defined by the maximum operating airspeed and the limits on normal load factor to prevent structural failure and injury to the pilot and passengers, as required by the Federal Aviation Regulations for airplane structural design. The normal load factor limits are -1.0g to 2.5g for flaps-up configuration and 0.0g to 2.0g for flaps-down configuration.

If any of these sub-envelopes are exceeded, the aircraft is considered to be in a flight envelope upset condition. A complete flight envelope recovery will therefore consist of high angular

rate recovery, aerodynamic envelope recovery (angle of attack and sideslip angle recovery), attitude envelope recovery (bank angle and pitch angle recovery), and flight vector recovery (flight path angle, underspeed and overspeed recovery). As a constraint, the normal load factor and maximum airspeed of the aircraft during the recovery must remain within the structural integrity envelope.

1.5 Proposed Strategy for Flight Envelope Recovery

A two-tier flight envelope recovery strategy is proposed that first performs high angular rate recovery and aerodynamic envelope recovery, and then performs attitude recovery, flight path angle recovery, and airspeed recovery.

The two-tier approach to flight envelope recovery was inspired by the conceptual split between fast rigid body rotational dynamics and the slow point mass translational dynamics argued by Peddle in his doctoral dissertation [5]. The rotational dynamics of the aircraft body axis system relative to the wind axis system operates over much shorter time scales than the rotational and translational dynamics of the wind axis system relative to the inertial axis system. The short period mode, roll mode, and Dutch roll mode of large transport aircraft, which represent the fast rotational dynamics, have much higher natural frequencies than the phugiod mode and the spiral mode, which represent the slow point mass translational dynamics. We therefore argue that angular rate and aerodynamic envelope recovery are performed through controlling the fast rigid body rotational dynamics, and that the attitude, flight path angle and airspeed recovery are performed through controlling the slower point mass translational dynamics. The time scale separation between the fast rotational dynamics and the slower point mass translational dynamics can be exploited to conceptually split the flight envelope recovery into an "inner loop recovery" of the angular rates and aerodynamic envelope, and an "outer loop recovery" of the attitude, flight path angle, and airspeed. Furthermore, we argue that angular rate and aerodynamic envelope recovery should be performed first, since it is a prerequisite for attitude, flight path angle and airspeed recovery.

The proposed strategy agrees with the pilot upset recovery procedures for large transport aircraft prescribed by the Airplane Upset Recovery Training Aid [6] and the pilot stall and spin recovery procedures for general aviation aircraft prescribed by the FAA Airplane Flying Handbook [7]. The pilot upset recovery procedures can be interpreted as executing the following sequence of actions: reduce the angular rates, recover the aerodynamic envelope, recover the attitude envelope, recover from overspeed, and finally recover altitude. We therefore propose that the flight envelope recovery be performed in stages, as shown in figure 1.1 and described in more detail below.

If the aircraft is experiencing high angular rates, they must be reduced before attempting to recover the aerodynamic envelope (angle of attack and sideslip angle). Moments are therefore required to damp the fast rotational dynamics of the aircraft. A passive approach would be to rely on the natural angular rate damping of the aircraft. However, if this is insufficient, then active rate damping should be added by feeding back the angular rate measurements to the control surfaces. Unfortunately, active rate damping may be complicated by nonlinear behaviour and/or model uncertainties in the forces and moments produced by the control surfaces in the post-stall region.

Assuming that the angular rates have been reduced to acceptable levels, the next step is to return the aircraft to its aerodynamic envelope, i.e. to recover the angle of attack and the sideslip angle, and to recover from underspeed. Forces are therefore required to recover the aircraft from underspeed and moments are required to control the angle of attack and the sideslip angle to return to the aerodynamic envelope. Again, a passive approach would be to rely on the natural dynamics of the aircraft.

The approach to underspeed recovery is to allow gravity to recover the airspeed, i.e. to

trade altitude for airspeed. This agrees with the prescribed pilot upset recovery procedure in the Airplane Upset Recovery Training Aid [6] which states that the goal of minimising altitude loss must be secondary to recovering from stall, and that flight crews must exchange altitude for airspeed. Unfortunately, this may not be an option for upsets conditions near the ground, since there may not be sufficient altitude to trade for airspeed.

The passive approach to recovering the angle of attack and the sideslip angle is to rely on the natural tendency of the aircraft to point its nose into the airflow. The danger is that the aircraft may have multiple stable equilibria, and may naturally be attracted to undesirable angles of attack and sideslip angles outside the aerodynamic envelope, depending on the initial conditions. To mitigate this risk, a bifurcation analysis should be performed to reveal alternative stable attractors. If alternative stable attractors exist, then the regions of attraction should be found to determine which initial conditions will stabilise to desirable equilibria inside the aerodynamic envelope, and which will stabilise to undesirable equilibria outside the aerodynamic envelope.

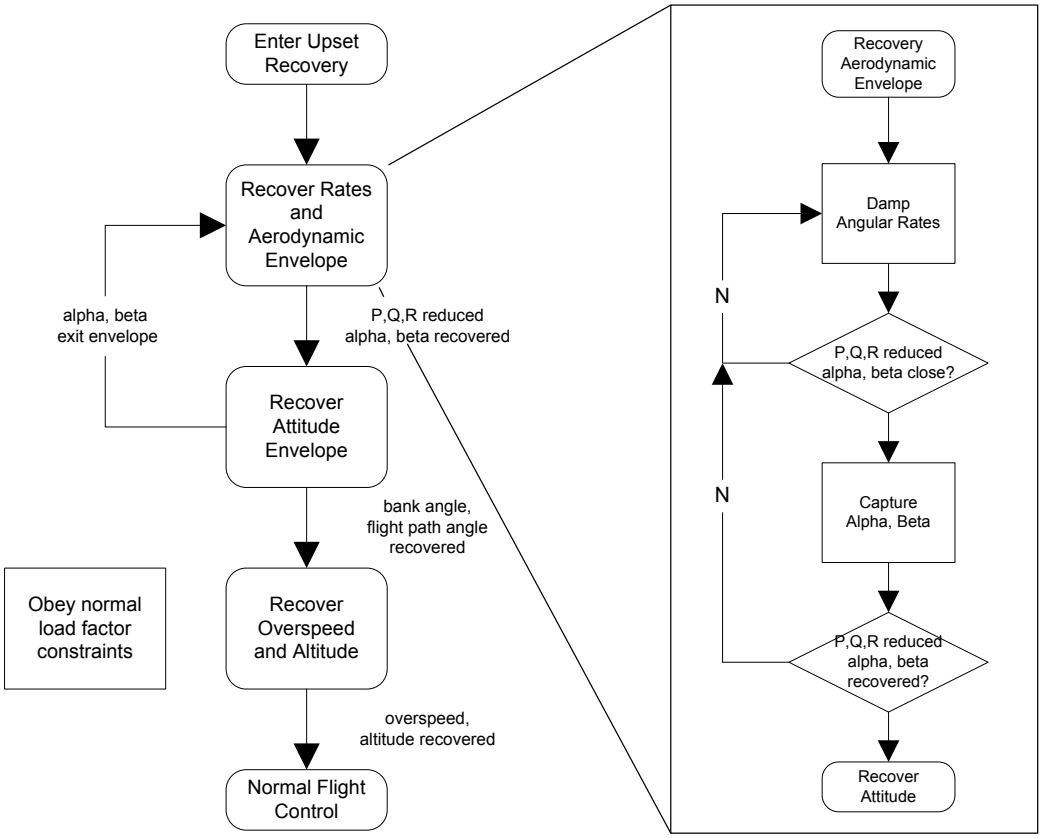


Figure 1.1: Flight envelope recovery flow chart

If the natural stability of the aircraft is insufficient, or if undesirable equilibria outside the aerodynamic envelope need to be avoided, then active angle of attack and sideslip angle control using the control surfaces will be required to steer the nose into the airstream. As with the active rate damping, the active aerodynamic envelope recovery is complicated by nonlinear behaviour and/or model uncertainties in the moments produced by the control surfaces in the post-stall region. Another challenge for active aerodynamic envelope recovery is that the angle of attack and sideslip angle may intermittently fall outside the valid sensor ranges of the anemometric sensors and may not be available for feedback.

Assuming that the aerodynamic envelope has been recovered, the next step is to return the aircraft to its attitude envelope, i.e. to recover the bank angle to wings level and to recover the flight path angle to level flight. (Recovering the flight path angle will implicitly also recover the pitch angle.) Since the airspeed, angle of attack and sideslip angle have been recovered by the previous step, the forces and moments produced by the control surfaces will behave linearly again, and the conventional flight control laws will be available again to perform the attitude and flight path angle recovery. An added benefit is that conventional flight control laws will provide load factor limiting to constrain the normal load factors to safe limits. The approach would therefore be to use roll rate control and bank angle control to return the bank angle to wings level, and to use normal load factor control and flight path angle control to return the flight path angle to level flight.

Once the attitude envelope has been recovered, it only remains to recover from overspeed and to regain altitude. The proposed approach is to reduce the throttle and to command a positive flight path angle to allow gravity to reduce the airspeed. Once the airspeed has been reduced to an acceptable range, the throttle may be increased again and a positive flight path angle may be commanded to regain altitude.

It should be noted that during all stages of recovery, the normal load factor must be constrained to safe limits as far as possible, especially during overspeed, to prevent structural failure and injury to the pilot and passengers.

1.6 Proposed Architectures for Flight Envelope Recovery

Two approaches for flight envelope recovery are proposed. The first approach uses a passive method to recover the angular rates and the aerodynamic envelope using the natural stability of the aircraft and then uses the conventional flight control laws, with all their protection functions, to recover the aircraft attitude, flight path angle and airspeed. The second approach uses an active method to recover the angular rates and the aerodynamic envelope using a Lyapunov-based inner-loop controller and uses an outer-loop controller based on optimal control theory to recover the aircraft attitude, flight path angle and airspeed while minimising the altitude loss.

For the passive aerodynamic envelope recovery, a bifurcation analysis is performed to identify the desirable stable equilibria of the aircraft inside the aerodynamic envelope, and to check for undesirable stable equilibria outside the aerodynamic envelope. The analysis reveals one desirable stable equilibrium branch inside the envelope, corresponding to stable trimmed symmetric flight, and also reveals two undesirable equilibria outside the flight envelope, corresponding to two stable post-stall spins in opposite directions.

For the conventional attitude and flight vector recovery, the conventional flight control laws with their protection functions are used to recover the attitude, flight path angle and airspeed of the aircraft. A bank angle controller with an inner-loop roll rate controller is used to return the bank angle to wings level, and a flight path angle controller with an inner-loop normal load factor controller is used to recover the flight path angle to level flight. A positive flight path angle is used to recover from overspeed using gravity. The angle of attack protection function prevents the aircraft from exiting the aerodynamic envelope again, and the load factor protection function keeps the normal load factor within safe limits.

For the active aerodynamic envelope recovery, a new Lyapunov-based controller is designed using a novel control Lyapunov function. The Lyapunov controller is not only able to recover the angular rates and the aerodynamic envelope, but also allows the angle of attack, sideslip angle, and wind axis roll rate to be actively controlled to track reference commands. The controller is designed to provide exponential stability of the aerodynamic recovery dynamics, and provides at least asymptotic stability when the control surfaces saturate.

For the optimal attitude and flight vector recovery, the problem is formulated as an optimal control problem where the objective is to find the optimal state trajectories and the optimal

sequence of control inputs that will recover the aircraft to straight and level flight with minimum altitude loss, while adhering to the aerodynamic and structural constraints of the aircraft. The optimal control problem is solved by applying a dynamic programming algorithm to a reduced order model of the aircraft's point mass translational dynamics.

The flight envelope recovery approaches are verified on the NASA Generic Transport Model (GTM), a wide-envelope aircraft model that is able to model the flight mechanics of large transport aircraft in out-of-envelope conditions.

1.7 Overview of Thesis

- In Chapter 1 the research topic was introduced. The motivation for the project was given, the research objectives were stated, and the primary contributions of the thesis were listed. The concept of flight envelope upset was defined, and an overview of the proposed flight envelope recovery strategy and architecture was given.
- In Chapter 2 a literature survey is performed to define the concepts of aircraft flight envelope and flight envelope upset, to survey the recovery techniques used by pilots to perform manual flight envelope recovery, to determine the state of the art of automatic flight envelope recovery functions, and to source wide-envelope aircraft models to use as the basis for the research.
- In Chapter 3 an overview is given of the NASA Generic Transport Model which is the wide-envelope aircraft model used as the basis for the research.
- Chapter 4 presents a passive approach to aerodynamic envelope recovery that uses the natural angular rate damping of the aircraft to recover from high angular rates, and uses the natural stability of the aircraft to recover the angle of attack and sideslip angle.
- Chapter 5 presents a state machine based approach to attitude and flight vector recovery that incorporates the passive aerodynamic envelope recovery as the first stage of the recovery. The state machine based approach illustrates that once the aerodynamic envelope has been recovered, the conventional flight control laws and protection functions can be used to recover the attitude, flight path angle, and airspeed.
- Chapter 6 presents an active aerodynamic envelope recovery method that employs a Lyapunov-based inner-loop controller to actively recover the angular rates, the angle of attack, and the sideslip angle from out-of-envelope conditions using the control surfaces of the aircraft.
- Chapter 7 presents an optimal attitude and flight vector recovery approach that simultaneously recovers the bank angle, flight path angle, and airspeed of the aircraft from flight envelope upset conditions while minimising the total altitude lost during the recovery maneuver.
- In Chapter 8, the inner-loop Lyapunov control law from chapter 6 and the outer-loop optimal attitude and flight vector recovery guidance law from chapter 7 are combined into a single integrated flight envelope recovery system and verified in simulation on the NASA Generic Transport Model.
- Chapter 9 presents a summary of the work done and gives recommendations for future research.

Chapter 2

Literature Survey

A literature survey was performed to define the concepts of aircraft flight envelope and flight envelope upset, to survey the recovery techniques used by pilots to perform manual flight envelope recovery, to determine the state of the art of automatic flight envelope recovery functions, and to source wide-envelope aircraft models to use as the basis for the design, analysis and verification of the automatic flight envelope recovery functions proposed in this thesis.

At the end of the chapter, the key conclusions from the literature survey are summarised, the research gaps are identified where research contributions can potentially be made, and the research decisions are made on how the research will be approached.

2.1 Flight Envelope

The flight envelope of an aircraft is the domain of flight conditions in which the aircraft can safely be operated without exceeding its aerodynamic and structural limits [4]. The flight envelope is often defined in terms of angle of attack and sideslip angle limits (the aerodynamic envelope), bank angle and pitch angle limits (the attitude envelope), and altitude, airspeed and normal load factor limits (the service flight envelope). Stated informally, the flight envelope represents the limiting boundaries within which the aircraft can be flown and recovered without exceptional pilot skill.

2.2 Definition of Flight Envelope Upset

The Boeing Company and the NASA Langley Research Center have jointly developed a set of metrics for defining Loss-of-Control through a NASA-funded partnership under the Aviation Safety and Security Program [1]. These metrics, collectively known as the Quantitative Loss-of-Control Criteria (QLC), are composed of five envelopes relating to the aircraft aerodynamics, flight dynamics, structural integrity and flight control use. The five QLC envelopes are shown in figure 2.1 and are listed below.

- the Adverse Aerodynamics envelope (angle of attack vs. sideslip angle)
- the Unusual Attitude envelope (bank angle vs. pitch angle)
- the Structural Integrity envelope (normal load factor vs. airspeed)
- the Dynamic Pitch Control envelope (dynamic pitch attitude vs. pitch control command)
- the Dynamic Roll Control envelope (dynamic roll attitude vs. roll control command)

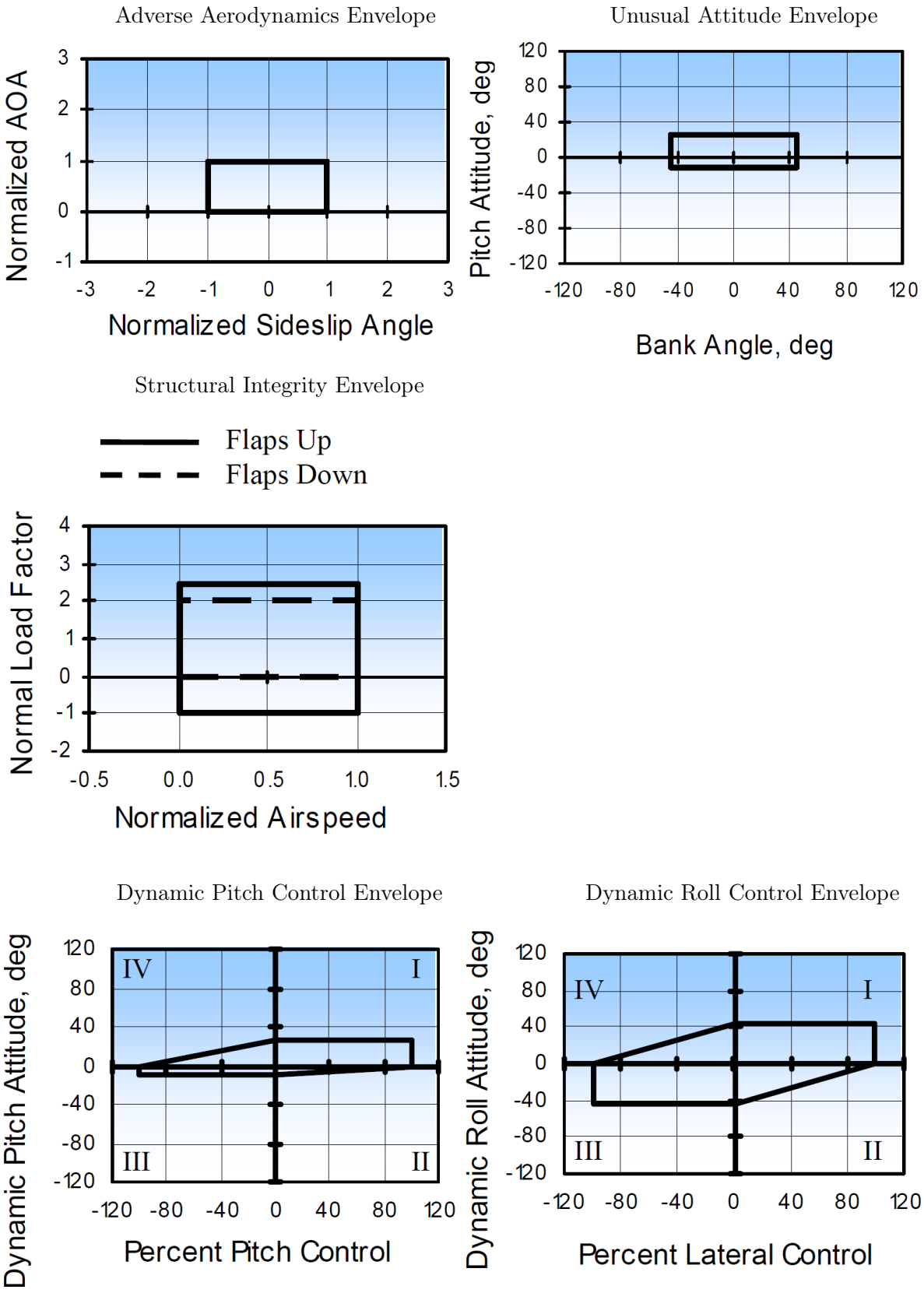


Figure 2.1: Quantitative loss-of-control envelopes defined by Wilborn et al [1]

The Adverse Aerodynamics envelope defines the angle of attack α and sideslip angle β limits for the given aircraft. Exceeding the angle of attack limits indicates that the aircraft has stalled, and exceeding the sideslip angle limits indicates large sideslip angles resulting in sideslip-induced roll.

The Unusual Attitude envelope defines the bank angle and pitch angle limits and is based on the generally accepted industry definition of "unusual attitude". Bank angles greater than 45 degrees (positive or negative), and pitch angles higher than 25 degrees nose up and lower than 10 degrees nose down, are considered to be unusual. Note that it is possible to violate the unusual attitude envelope without violating the adverse aerodynamics envelope, and vice versa.

The Structural Integrity envelope defines the airspeed vs normal load factor limits of the aircraft. The normal load factor limits are based on the Federal Aviation Regulation requirements for aircraft structural design. The prescribed normal load factor limits are -1 to +2.5 g for flaps-up configuration and 0 to +2 g for flaps-down configuration. Exceeding the Structural Integrity envelope indicates accelerated stall, overspeed, and structural overload.

The Dynamic Pitch Control envelope maps the pitch axis control authority against the dynamic pitch attitude. The dynamic pitch attitude is the sum of the current pitch attitude and its expected change over one second. The dynamic pitch envelope indicates whether there is sufficient pitch control authority to arrest the pitching motion before the pitch attitude exceeds the unusual attitude envelope.

The Dynamic Roll Control envelope maps the roll axis control authority against the dynamic roll attitude. The dynamic roll attitude is the sum of the current roll attitude and its expected change over one second. The dynamic roll envelope indicates whether there is sufficient roll control authority to arrest the rolling motion before the roll attitude exceeds the unusual attitude envelope.

Although these Quantitative Loss-of-Control Criteria envelopes are useful for classifying accident data as Loss-of-Control and to suggest which variables are involved, they do not connect directly to the flight mechanics of the aircraft. However, they do imply that flight envelope recovery should address high angular rate recovery, aerodynamic envelope recovery, attitude recovery, airspeed recovery and normal load factor recovery.

The flight envelopes conceptualised by the author (in Chapter 1, Section 1.4), namely the Aerodynamic Envelope, the Attitude Envelope, the Flight Vector Envelope and the Structural Integrity Envelope, are re-interpretations of the QLC envelopes proposed by Wilborn et al, and more directly represents the objectives of flight envelope recovery.

2.3 Types of Flight Envelope Upsets

This section provides an overview of different types of flight envelope upset conditions found in the literature. The definitions and explanations provided below are drawn from the FAA Airplane Flying Handbook [7], from a technical report on the *Aerodynamic Characteristics of Airplanes at High Angles of Attack* by Chambers and Grafton [8], and from a technical report on the *Fundamentals and Methods of High Angle of Attack Flying Qualities Research* by Seltzer and Rhodeside [9].

2.3.1 Stall

Stall is the condition where the angle of attack exceeds the stall angle of attack. The stall angle of attack is the angle of attack associated with the maximum usable lift for a given flight condition. At angles of attack below the stall angle of attack, the lift increases proportionally (and quite linearly) with an increase in angle of attack. When the angle of attack exceeds the stall angle of attack, the lift decreases with an increase of angle of attack, instead of increasing.

It is important to note that stall is only a function of angle of attack, and is independent of airspeed, gross attitude (pitch angle and bank angle), and thrust setting. It is therefore possible for an aircraft to have an unusual attitude (high nose-up or nose-down pitch angle and/or high bank angle) and not be stalled. Conversely, it is possible for an aircraft to be stalled, even if it has a normal airspeed or a normal attitude (nose level and wings level). The stall condition is determined by the orientation of the aircraft body relative to its velocity vector, and not by the orientation of the aircraft body relative to the Earth.

The stall angle of attack is typically about 15 degrees, while the angle of attack during normal cruise flight typically varies between about 3 and 7 degrees.

2.3.2 Post-Stall

Post-stall is the flight regime involving angles of attack greater than the stall angle of attack. The aircraft characteristics in the post-stall regime may consist of several distinct types of aircraft motion, namely departure, post-stall gyration, spin, and deep stall.

2.3.3 Departure

Departure is the event in the post-stall flight regime which precipitates entry into a post-stall gyration, spin, or deep stall condition. The departure may be characterised by divergent, large amplitude, uncommanded aircraft motions, such as nose-slice or pitch-up. Nose-slice is an uncommanded lateral-directional motion viewed by the pilot as a divergence in yaw. Pitch-up is an uncommanded, sudden increase in angle of attack.

2.3.4 Post Stall Gyration

Post stall gyration is an uncontrolled motion about one or more aircraft axes following departure. While this type of airplane motion involves angles of attack higher than the stall angle, lower angles may be encountered intermittently. A post-stall gyration is distinguished from a spin by a lack of a predominant, sustained yawing motion, and by the potential for exhibiting sub-stall angles of attack.

2.3.5 Spin

Spin is a steep spiral motion with the angle of attack above the stall angle and with the aircraft descending rapidly while rotating about a vertical axis. The aircraft autorotates around the vertical axis due to the rising wing being less stalled than the descending wing, creating a simultaneous rolling, yawing, and pitching motion with the aircraft at high angle of attack and sideslip. A spin is caused when the aircraft's wing exceeds the stall angle of attack with a sideslip acting on the aircraft. The spin is considered to have four phases: spin entry, incipient spin, developed spin, and spin recovery.

Spin Entry: Spin entry from unstalled flight may be deliberate or inadvertent. A deliberate spin is initiated by slowing the aircraft towards stall and then at the point of stall generating a yaw rate by applying full rudder deflection. The yawing motion promotes stalling on the rearward travelling wing due to increased angle of attack while maintaining attached airflow on the forward travelling wing due to reduced angle of attack. The rearward travelling wing experiences a large loss of lift, while the forward travelling wing maintains its lift. The resulting differential lift produces a rolling moment in the direction of the rearward travelling wing, and initiates the spin with a large roll rate.

Incipient Spin: The incipient spin is the transition between the spin entry and the developed spin. During the incipient spin, the aircraft flight path angle changes from horizontal to vertical or nearly vertical, the angle of attack increases to well beyond the stall angle of attack,

and the yaw rate increases to match or exceed the roll rate. The incipient phase is considered to end when the airspeed has become steady and a vertical trajectory has been reached. For practical purposes, the developed spin is reasonably well established after two to three yaw rotations.

Developed Spin: In developed spin, the aircraft describes a steep spiral motion about a vertical axis, in which the spin rate, angle of attack, sideslip angle, and airspeed are constant. In many cases, the motion does not reach a steady equilibrium state, but may exhibit an oscillation about the nominal equilibrium state, with a frequency higher than the spin rate.

The developed spin may be described as a "steep spin" or a "flat spin" based on the angle of attack of the aircraft during the spinning motion. The following guidelines have been provided by NASA to define classical spin modes observed in spinning aircraft

Table 2.1: NASA spin mode classifications.

Spin Mode	Angle of Attack	Nose-Down Attitude
Steep	20 - 30°	60 - 70°
Moderately Steep	30 - 45°	45 - 60°
Moderately Flat	45 - 65°	25 - 45°
Flat	65 - 90°	0 - 25°

For example, an aircraft spinning at an angle of attack of 75° is considered to be in a flat spin. The term "flat spin" derives from the fact that the pitch attitude of the aircraft is close to horizontal ("flat") while the aircraft is descending nearly vertically.

Spin Recovery: The spin recovery occurs when the angles of attack of the wings decrease below the stall angle of attack and the autorotation slows. The aircraft then recovers from stall and stops spinning.

2.3.6 Deep Stall

Deep stall is a stall at high angles of attack well beyond the stall angle of attack where nose-down pitching moment cannot be generated with the application of full nose-down elevator, making stall recovery difficult or impossible. Deep stall is most often exhibited by aircraft with high horizontal tailplanes, such as jet fighter aircraft with high horizontal tails and T-tail transport aircraft. At very high angles of attack the horizontal tail is immersed in the combined wake of the wings, engines and fuselage. The effect is to produce a nonlinear variation of pitching moment as a function of angle of attack, resulting in two ranges of stable trim points, the normal range at low angle of attack, and another range at high angles of attack well beyond stall. The effectiveness of the elevators is also significantly decreased due to the low-energy wake impinging on the horizontal tailplane, leaving insufficient nose-down pitching moment to recover from the deep stall trim point.

2.3.7 Inertial Coupling

Inertial coupling is a phenomenon observed in high-speed flight of jet fighter aircraft where high roll rates result in uncommanded pitching and yawing motion, due to the relatively low roll moment of inertia relative to the pitch and yaw moments of inertia. The phenomenon is not aerodynamic, but is caused by the inertial properties of the aircraft. The aircraft may be visualised as a long slender rod with two end masses. If the rod is spun about a rotation axis that makes an angle relative to the rod axis, the centrifugal forces acting on the two end masses will cause the rod to tilt outwards relative to the rotation axis.

2.3.8 Unusual Pitch Angle / Unusual Flight Path Angle

A large pitch angle (nose up or nose down) is considered an out-of-envelope condition because it implies either that the angle of attack is out-of-envelope, or that the flight path angle is out-of-envelope. Note that the pitch angle is the angle between the longitudinal body axis of the aircraft and the horizontal plane, while the flight path angle is the angle between the velocity vector of the aircraft and the horizontal plane. If the flight path angle is level, then a large nose-up pitch angle implies that the angle of attack is beyond the stall angle of attack. If the angle of attack is inside the aerodynamic envelope, then a large nose-up pitch angle implies that the aircraft has a steeply climbing flight path angle, while a large nose-down pitch angle implies a steeply descending flight path angle.

A steeply ascending flight path angle is considered an out-of-envelope condition, because it will eventually lead to an underspeed condition if their is insufficient thrust available to maintain the climb. A steeply descending flight path angle is considered an out-of-envelope condition, since it will eventually lead to an overspeed condition or impact with the ground.

2.3.9 High or Inverted Bank Angle

A high or inverted bank angle is considered an out-of-envelope condition because it means that the directon of the aircraft lift vector is oriented either horizontally or downwards and is therefore not in the correct orientation to counter the gravitational force.

2.3.10 Underspeed

The aircraft is considered to be in an underspeed condition if the airspeed is below the stall speed. As stated before, stall depends only on the angle of attack, and not on the airspeed of the aircraft. However, the slower an aircraft flies, the more angle of attack it needs to produce lift equal to the weight of the aircraft. The stall speed is the airspeed where the angle of attack required to produce lift equal to the weight of the aircraft equals the stall angle of attack. At airspeeds below the stall speed, there is no angle of attack that would produce enough lift to maintain level flight.

2.3.11 Overspeed

The aircraft is considered to be in an overspeed condition if the airspeed exceeds its maximum operating limit speed V_{MO}/M_{MO}. The absolute maximum speed above which the aircraft must not fly is called the dive speed V_D. To achieve this speed, the aircraft must enter a dive (steep descent), as the engines cannot produce sufficient thrust to overcome aerodynamic drag in level flight. At the dive speed, excessive aircraft vibrations develop which put the aircraft structural integrity at risk.

2.3.12 Excessive Load Factor

The aircraft is considered to exceed the structural integrity envelope if its load factor exceeds the maximum load factor limits of the aircraft. The load factor is defined as the ratio of the lift of the aircraft to its weight. The load factor is a measure of the forces to which the aircraft structure and the passengers are subjected.

Since the load factor is a ratio of two forces, it is dimensionless. However, the load factor is commonly expressed in g units, due to the fact that an observer on board the aircraft will experience an apparent gravitational acceleration equal to the load factor multiplied by the standard gravitational acceleration. For example, when an aircraft performs a maneuver with a load factor of 2, the expression "the aircraft is pulling 2 g's" is often used.

2.4 Manual Flight Envelope Recovery

This section provides an overview of prescribed pilot flight envelope recovery procedures. The Airplane Upset Recovery Training Aid [6] provides recommended procedures for stall recovery, unusual pitch angle recovery, and bank angle recovery, for large transport aircraft. The FAA Airplane Flying Handbook [7] provides recommended spin recovery procedures.

2.4.1 Stall Recovery

The Airplane Upset Recovery Training Aid recommends the following procedure for stall recovery. First, the pilot must recognise and confirm the stall condition. The airplane is stalled when the angle of attack is greater than the stall angle of attack. The stall is characterised by any of, or a combination of, the following: buffeting, a lack of pitch authority, a lack of roll control, and/or an inability to arrest the descent rate. The stall characteristics are usually also accompanied by a continuous stall warning.

To recover from stall, the angle of attack must be reduced to below the stalling angle by applying nose-down pitch control and maintaining it until the aircraft is recovered from stall. For aircraft with underwing-mounted engines, it may be necessary under certain conditions to reduce the thrust to prevent the angle of attack from continuing to increase.

The procedure emphasizes that if the aircraft is stalled, it must first be recovered from the stalled condition before initiating further upset recovery procedures.

2.4.2 Pitch Angle Recovery: Nose High

The Airplane Upset Recovery Training Aid recommends the following procedure for nose-high, wings-level pitch angle recovery. The aircraft is assumed to be an upset situation with a pitch angle greater then 25 degrees nose high, but with its wings level. The airspeed is decreasing rapidly, and the ability to maneuver is also decreasing.

Nose-down elevator should be applied to achieve a nose-down pitch rate. If the pitch rate is not immediately under control, there are several other techniques that may be tried. For aircraft with underwing-mounted engines, a nose-down pitch rate may be achieved by reducing the thrust, if altitude permits. The pitch rate may also be controlled by rolling the airplane to a bank angle that starts a nose-down pitch rate. This bank angle should normally not exceed about 60 degrees. If the control provided by the ailerons and spoilers is ineffective, rudder input may be required to induce a rolling maneuver for recovery. To complete the recovery, the aircraft must be rolled to wings-level, if necessary, as the nose approaches the horizon. The aircraft should be recovered to a slightly nose-low attitude to reduce the potential for entering another upset.

2.4.3 Pitch Angle Recovery: Nose Low

The Airplane Upset Recovery Training Aid recommends the following procedures for nose-low, wings-level pitch angle recovery. The aircraft is assumed to be an upset situation with a pitch angle greater then 10 degrees nose low, but with its wings level. Two different procedures are provided: one for nose-low, low-airspeed situations, and another for nose-low, high-airspeed situations.

Nose Low, Airspeed Low

In a nose-low, low-airspeed situation, the aircraft may be stalled even at a relatively low pitch angle. If the aircraft is stalled, it must be recovered from the stall first before proceeding with the pitch angle recovery. The stall recovery may require nose-down elevator to be applied, which is counter-intuitive when also recovering from a nose-low pitch angle. Once the aircraft is

recovered from stall, thrust must be applied to recover the airspeed and nose-up elevator must be applied to return the nose to the desired pitch angle. While applying nose-up elevator, a secondary stall must be avoided. The aircraft load factor and airspeed limitations must also be respected during the pitch angle recovery.

Nose Low, Airspeed High

In a nose-low, high-airspeed situation, nose-up elevator must be applied to recover the pitch angle from the nose-low upset situation. For extreme out-of-trim conditions, it may also be necessary to cautiously apply stabiliser trim to assist in obtaining the desired nose-up pitch rate. The thrust must be reduced and, if necessary, the speedbrakes must be extended to recover from the high airspeed. When applying nose-up elevator, the pilot must apply enough nose-up elevator to avoid impact with the terrain, but not so much that the aircraft enters an accelerated stall by exceeding the stall angle of attack. The aircraft load factor and airspeed limitations must be respected during the pitch angle recovery.

2.4.4 Bank Angle Recovery

The Airplane Upset Recovery Training Aid recommends the following procedure for bank angle recovery. The aircraft is assumed to be an upset situation with a bank angle greater then 45 degrees, and may even exceed 90 degrees. The prescribed recovery procedure states that in high-bank-angle situations, the primary objective is to roll the airplane in the shortest direction to near wings level. However, if the airplane is stalled, it is first necessary to recover from the stall.

Two different procedures are provided: one for high-bank-angle, nose-high situations, and another for high-bank-angle, nose-low situations.

High Bank Angle, Nose High

The aircraft is assumed to be in an upset situation with a bank angle greater than 45 degrees, and a pitch angle greater than 25 degrees, nose high. The airspeed is assumed to be decreasing.

In high-bank-angle, nose-high situations, a large bank angle is actually helpful in reducing the high nose-up pitch angle. The angle of attack must be reduced and the bank angle must be adjusted not to exceed 60 degrees, in order to achieve a nose-down pitch rate. The pilot should maintain awareness of the energy management and aircraft roll rate. To complete the recovery, the aircraft must be rolled to wings level as the nose approaches the horizon.

High Bank Angle, Nose Low

The aircraft is assumed to be in an upset situation with a bank angle greater than 45 degrees, and a pitch angle exceeding 10 degrees nose low. The airspeed is assumed to be increasing. A nose-low, high-angle-of-bank attitude requires prompt action, because altitude is rapidly being exchanged for airspeed. Even if the airplane is at an altitude where ground impact is not an immediate concern, airspeed can rapidly increase beyond the aircraft design limits.

Roll control must be applied to roll the aircraft to wings level. It may also be necessary to simultaneously adjust the thrust to avoid overspeed. The angle of attack may also have to be reduced (by reducing nose-up elevator) to improve the effectiveness of the roll control and to avoid pointing the lift vector towards the ground. Full aileron and spoiler input may be necessary to establish a recovery roll rate towards the nearest horizon. It is important that nose-up load factor not be increased and that nose-up elevator or stabilizer trim not be used until the airplane approaches wings level. If the application of full ailerons and spoilers is not satisfactory, it may be necessary to apply a small amount of rudder in the direction of

the desired roll. Too much rudder applied too quickly or held too long may result in loss of lateral and directional control and cause structural damage. As the wings approach level, the speedbrakes must be extended to recover the airspeed, if required.

For large transport-category aircraft, the pilot should not attempt to roll through (add pro-roll controls) during an upset in order to achieve wings level more quickly, but should roll in the shortest direction to wings level.

2.4.5 Spin Recovery

The FAA Airplane Flying Handbook provides the following procedure for spin recovery. The spin recovery is achieved by applying control inputs that disrupt the spin equilibrium by stopping the rotation and recovering from the stall. The handbook that states that the aircraft manufacturer's recommended spin recovery procedures should be followed. However, in the absence of the aircraft manufacturer's spin recovery procedures and techniques, the following general spin recovery procedure is recommended:

1. Reduced the throttle to idle. The thrust aggravates the spin, and usually results in a flatter spin attitude and increased rotation rates.

2. Positioned the ailerons to neutral. The ailerons may have an adverse effect on spin recovery. Aileron control in the direction of the spin may speed up the rate of rotation and delay the recovery. Aileron control opposite the direction of spin may cause the down aileron to move the wing deeper into the stall and aggravate the situation. The best procedure is to ensure that the ailerons are neutral.

3. Apply full opposite rudder against the rotation.

4. Apply nose-down elevator to reduce the angle of attack and recover the aircraft from stall. This should be done immediately after full rudder application. The nose-down elevator must be maintained until the stall recovery is achieved. When the aircraft is recovered from stall, the spinning will stop.

5. After the spin rotation stops, neutralise the rudder. If the rudder is not neutralized, the increasing airspeed acting upon the deflected rudder will cause a yawing and skidding effect.

6. Begin applying nose-up elevator to raise the nose to level flight. Caution must be used not to apply excessive nose-up elevator after the rotation stops, since it can cause a secondary stall and result in another spin. Care should be taken not to exceed the load factor and airspeed limits of the aircraft during the spin recovery.

2.5 Automatic Flight Envelope Recovery

This section provides an overview of automatic flight envelope recovery research found in the open literature. Research on flight envelope protection and extension is also relevant and is therefore also included. The majority of research found on flight envelope extension and recovery is aimed at high performance fighter aircraft, rather than large transport aircraft. Although techniques applicable to fighter aircraft may not necessarily be applicable to large transport aircraft, the research is also included in the literature survey.

Some research on aircraft upset recovery is dedicated to handling sensor failures, actuator failures and airframe damage, and not specifically to recovering from flight envelope upset conditions. A large body of research also exists on fault detection and isolation and fault-tolerant control for aircraft. Since the research presented in this thesis assumes that the aircraft

is healthy and fully functional, but just finds itself in a flight envelope upset condition, this research will be omitted from the literature survey.

2.5.1 Flight Envelope Protection on Modern Commercial Airliners

Many modern commercial airliners include flight envelope protection functions as part of their fly-by-wire flight control systems. The flight envelope protection system prevents the pilot from giving control commands that would force the aircraft to exceed its flight envelope. The architecture and design of the fly-by-wire flight control system and the flight envelope protection functions are proprietary to the manufacturers of large commercial airliners, and not much information is available in the open literature. However, some information was obtained from an article by Favre [10] that presented an overview of the Airbus fly-by-wire control laws used on the Airbus A320 and the Airbus A340.

The flight envelope protection functions integrated into the longitudinal control law include

- normal load factor limiting
- angle of attack protection
- underspeed protection
- overspeed protection
- pitch attitude protection
- flight path angle protection (for autopilot only)

The flight envelope protection functions integrated into the lateral control law include

- roll rate limiting
- bank angle protection

It should be noted that these envelope protection functions are only designed to prevent the aircraft from exiting the flight envelope, and not for recovering the flight envelope after it has already exited.

2.5.2 Flight Envelope Extension and Protection

The following research on flight envelope extension and protection was found in the literature:

Li et al [11] presented an on-line learning flight control scheme that uses radial basis function neural networks to adapt to changes in the aircraft dynamics and provide good performance for aircraft undergoing highly nonlinear maneuvers. The neural controller was demonstrated on the simulation model of a high performance fighter aircraft executing a high angle-of-attack roll manuever.

Richardson et al [12] presented an approach to controlling nonlinear aircraft dynamics over a wide range of parameters by incorporating bifurcation analysis and continuation methods into the controller design process. The approach uses feedforward control to generate a set of equilibria for which a linear relationship exists between the commanded angle-of-attack and the steady-state angle of attack. Feedback control is then used to stabilise the dynamics about the equilibrium branches created by the feedforward control. The continuation based control was demonstrated on a simulation model of the Hypothetical High Incidence Research Model (HHIRM).

Sibilski [13] used continuation methods and bifurcation analysis to identify control law parameters for post-stall manoeuvring of fighter aircraft. The continuation method was used

to determine the steady states of a thrust-vectoring, supersonic combat aircraft with a streaked wing, as functions of the deflections of thrust vector, and to predict instabilities in the post-stall regime.

Paranjape and Ananthkrishnan [14] used the extended bifurcation and continuation method to simultaneously analyse the performance and stability of a fighter aircraft in a maximum turn rate manuever.

Wu et al [15] performed aerodynamic modelling and gain-scheduled control design for fighter aircraft flight in near- and post-stall conditions. Adequate control capability was maintained with stability and performance guaranteed.

Jones et al [16] applied tailored dynamic gain-scheduled control to a highly nonlinear unmanned combat air vehicle model. The method uses continuation tailoring and multi-variable gain-scheduling to schedule the control gains with a fast-varying parameter rather than with a slow-varying parameter.

Nguyen et al [17] investigated adaptive control methods for stability recovery of damaged aircraft operating in off-nominal flight conditions under damage and/or failures. Aerodynamic modelling was performed to assess changes in the stability and control derivatives due to damage to the wings, horizontal tailplane, and vertical tailplane of the NASA Generic Transport Model (GTM). Simulations showed that a direct adaptive control law was able to stabilise the damaged aircraft.

Richardson et al [18] developed a novel technique using continuation methods in which surfaces of equilibria were formed for a highly nonlinear aircraft dynamic system with a feedforward- and feedback-scheduled controller. The method allows the creation and destruction of equilibria in the controlled system to be visualised as the control gains are varied. The technique was demonstrated on the Hypothetical High Incidence Research Model (HHIRM).

Shin and Gregory [19] applied robust gain-scheduled control using a linear parameter-varying control synthesis method to design fault tolerant control for a civil transport aircraft. Passive and active fault-tolerant controllers were designed for the longitudinal dynamics of the Boeing 747-100/200 in the presence of elevator failure.

Shin and Belcastro [20] described an analysis method for determining a reliable flight regime in the flight envelope within which a integrated resilient control system can achieve desired performance. The method was demonstrated on linear fractional transform models of the Boeing 747-100/200.

Gregory et al [21] applied L1 adaptive control to implement envelope protection for the NASA Generic Transport Model under adverse flight conditions including unusual attitudes, surface failures and structural damage. The objective of the research was that the adaptive controller should learn fast enough to keep the aircraft within the extended envelope.

Kwatny et al [22] considered strategies for envelope protection and envelope restoration for both unimpaired and impaired aircraft based on safe set theory. Examples were given using a simplified fourth order model of the longitudinal dynamics of the NASA GTM. Envelope protection was implemented using a stabilising, discontinuous switching control law. Safe set theory was used to determine the set of admissible controls that will prevent departure from the flight envelope.

2.5.3 Flight Envelope Recovery

The following research on flight envelope recovery was found in the literature:

Combs et al [23] described the pilot activated automatic recovery system (PAARS) on the F-117A stealth aircraft. The system provides an all attitude recovery capability to the aircraft should the pilot become disoriented due to sensor / display failures or due to distraction brought about by excessive pilot workload. The system is engaged by the pilot by depressing a button on the control stick grip. The design and operation of the PAARS mode is based on providing

automated responses to non-nominal flight attitudes similar to that which a fully lucid pilot would perform for recovery. The PAARS control laws perform pitch recovery, roll recovery and speed recovery by executing a specific sequence of actions after classifying the initial attitude of the aircraft into one of seven sectors. The main weakness of this approach is that it is only designed for attitude recovery, and does not provide for aerodynamic envelope recovery (spin recovery or recovery from the post-stall regime).

Saraf et al [24] designed a feedback linearisation based nonlinear controller to recover an unstable aircraft from the post-stall regime. The nonlinear controller is engaged after the aircraft departs from the normal flight regime. The controller stabilises the aircraft in a stable spin. Then a set of synthetic pilot inputs are applied to suppress yaw rate and reduce angle of attack to cause an automatic transition from the spin equilibrium to low angles of attack where a second controller is engaged. The second controller is a normal gain-scheduled controller that is designed to have a large domain of attraction at low angles of attack. The second controller traps the aircraft into a low angle of attack level flight. The recovery controller was verified on a six degree of freedom nonlinear simulation. The main weakness of the approach is that the synthetic pilot inputs that cause the transition from the high angle of attack spin equilibrium to low angles of attack are very specific to the aircraft model. The paper presents no general method for determining these synthetic inputs, beyond an extensive trial and error study. The feedback linearisation design is also very dependent on the aircraft model, and is not robust to variations in the aerodynamic derivatives, mass, or moment of inertia of the aircraft.

Sparks and Moerder [25] investigated optimal aircraft upset recovery with and without component failures using a sequential quadratic programming algorithm.

Kumar et al [26] used bifurcation techniques together with nonlinear dynamic inversion for spin prediction and recovery of a fighter aircraft. A bifurcation analysis was performed to identify possible spin states, and then a nonlinear dynamic inversion controller was implemented for recovery from an oscillatory spin to a level flight trim. The method was demonstrated in simulation on the High Angle-of-attack Research Vehicle (HARV) model.

Lee and Nagata [27] presented a momentum vector control law for spin recovery. The directions and magnitudes of linear and angular vectors were effected to restore prespin conditions. The computation of the required controls was formulated as a nonlinear programming problem.

Raghavendra et al [28] presented a spin recovery control law, with and without thrust vectoring, using nonlinear dynamic inversion. The disadvantage of the approach is that the use of thrust vectoring is only applicable to fighter aircraft and not to large transport aircraft.

Dutoi et al [29] presented a spin recovery system that combines robust control and reinforcement learning. Off-line reinforcement learning techniques are applied to simulation data to discover recovery strategies that improve on known strategies. When learning is complete, the strategies are provided to an online component. In the event of an upset, the online component is interrogated to determine the best control decision at each control update until the recovery is complete. The system is partitioned into two components; one which focusses on recovery from high angular rate upsets and one another which focusses on recovery from unusual attitude upsets. The approach was demonstrated in simulation using NASA's Generic Transport Model (GTM). The disadvantage of the approach is that it requires known upset recovery strategies as a starting point, and only improves on these strategies.

Sinha and Rao [30] presented a sliding-mode controller for spin recovery based on a variable-structure control technique. The sliding-mode controller was tested in simulation using the F-18/HARV model, which is a fighter aircraft model with highly nonlinear post-stall dynamics. However, sliding mode control uses very aggressive control actions (high control gains and rapid switching of control surfaces) to force the trajectory of the aircraft to slide along a designed sliding manifold. This may lead to actuator chatter, aircraft structural damage, and excitation of unmodelled aircraft structural dynamics. The high control forces and moments required by the sliding mode control are also more suited to agile fighter-type aircraft than to sluggish

transport aircraft.

Dongmo presented a loss-of-control recovery controllers using feedback linearisation and high order sliding mode control [31], as well as optimal high order sliding mode control with discontinuous observers [32]. The controllers were tested in simulaton on the NASA GTM. However, the test scenarios did not include high angle of attack upsets, and therefore the aircraft did not actually leave the aerodynamic envelope.

Crespo et al [33] designed several upset recovery control strategies, consisting of control laws and command laws, and evaluated and compared them on the NASA GTM. The control strategies considered were fixed control deflections, fixed-gain feedback controllers with constant commands, fixed-gain feedback controllers with state-varying commands, and two gain-scheduled feedback controllers with state-varying commands. However, the purpose of the paper was rather to present a framework for the systematic evaluation of recovery strategies, and not to design the control strategies themselves. It should also be noted that multi-mode or state machine based recovery strategies were not considered in the paper.

Kwatny et al [22] considered strategies for envelope protection and envelope restoration for both unimpaired and impaired aircraft based on safe set theory. Examples were given using a simplified fourth order model of the longitudinal dynamics of the NASA GTM. Envelope recovery was implemented using a simple linear quadratic regulator. Safe set theory was used to determine the recoverable set of initial conditions. However, the technique is computationally expensive and is currently only suited for lower order models with limited practical application.

2.6 Wide Envelope Aircraft Models

A wide-envelope aircraft model is required to serve as the basis of the flight envelope recovery research. Currently, four aircraft models are widely used for upset recovery research, of which three are for fighter aircraft, and only one is for large commercial airliners. The three-wide envelope fighter aircraft models are the Hypothetical High-Incidence Research Model (HHIRM), the Aero-Data Model in a Research Environment (ADMIRE), and the High Angle-of-attack Research Vehicle (HARV). The wide-envelope commercial airliner model is the NASA Generic Transport Model (GTM). Recently, another wide-envelope commercial airliner model was developed under a European Union 7th Framework Program project called SUPRA (Simulation of Upset Recovery in Aviation).

2.6.1 The Hypothetical High-Incidence Research Model (HHIRM)

The Hypothetical High Incidence Research Model (HHIRM) [34–36] was created in order to provide a benchmark for nonlinear aircraft dynamics analysis and control design. It was developed by Goman et al in 1995 for the then UK Defense and Research Agency (DRA). The aerodynamic characteristics of the HHIRM are very similar to those of many existing combat aircraft and can be easily tuned to arbitrarily chosen characteristics for qualitative nonlinear dynamics analysis and control law design.

2.6.2 The Aero-Data Model in a Research Environment (ADMIRE)

The Aero-Data Model in a Research Environment (ADMIRE) [37] is a non-linear, six degree of freedom simulation model of a rigid small fighter aircraft with a delta-canard configuration. The development of ADMIRE was started at The Aeronautical Research Institute of Sweden in 1997 on a contract from the Swedish Defence Materiel Administration. The goal was to use the Generic Aerodata Model (GAM), developed by Saab AB and The Swedish Royal Institute of Technology, as a basis and construct a complete aircraft model for research use that can be freely distributed. The original Generic Aerodata Model has been complemented with a

flight control system and models of the engine, dynamics, and actuators. ADMIRE is currently maintained by FOI, Sweden.

2.6.3 The High Angle-of-attack Research Vehicle (HARV)

The High Angle-of-attack Research Vehicle (HARV) [38] was an F-18 Hornet fighter aircraft used by NASA's Dryden Flight Research Center in a three-phased flight research program lasting from April 1987 until September 1996. The aircraft demonstrated stabilized flight at angles of attack between 65 and 70 degrees using thrust vectoring vanes, a research flight control system, and eventually forebody strakes. Forebody strakes are hinged structures on the forward side of the fuselage that provide control by interacting with vortices generated at high angles of attack to create side forces. The combination of technologies provided carefree handling of a fighter aircraft in a part of the flight regime that was otherwise very dangerous. Flight research with the HARV increased the understanding of flight at high angles of attack and enabled designers of fighter aircraft to design airplanes that fly safely in portions of the flight envelope that pilots previously had to avoid.

2.6.4 The Generic Transport Model (GTM)

The Generic Transport Model (GTM) is a 5.5% dynamically scaled unmanned aerial vehicle (UAV) developed by NASA for experimental flight tests outside the normal flight envelope that are considered too risky for full-scale manned transport airplane testing [39].

Aerodynamic models were developed for the GTM based on wind tunnel tests conducted at the NASA Langley Research Center using a model that has the same mold line specifications [2, 40, 41]. Steady data were obtained for different configurations and an extensive range of angles of attack, sideslip angles, and control surface deflections [42]. Angular rate effects were modeled by blending data sets obtained from oscillatory motion rigs (i.e. forced oscillation) and steady motion rigs (i.e. rotary balance) [43]. Aerodynamic modeling of airframe damage effects was also performed using an aerodynamic code to assess changes in the stability and control derivatives of the GTM [44]. From these aerodynamic models, a full non-linear flight dynamics simulation model of the GTM was created and made available by NASA for research use. The NASA GTM serves as a common research model and is used internationally by researchers in both academia and industry for research on the nonlinear flight dynamics and upset recovery of large transport aircraft.

2.6.5 Simulation of Upset Recovery in Aviation (SUPRA)

The SUPRA research project (Simulation of Upset Recovery in Aviation) was a European Union 7th Framework Program [45, 46]. One of the primary objectives of the SUPRA project was to develop and validate an extended aerodynamic model of a generic large transport airplane to be used for piloted simulation in the post-stall region and for upset recovery training. The aerodynamic model was created from experimental wind tunnel data obtained for an airliner with two under-wing mounted engines and a conventional tail. The aerodynamic data were obtained in low speed and transonic wind tunnels. Additional aerodynamic data were generated using computational fluid dynamics methods. The basic aircraft geometry for the CFD study was a T-tail aircraft configuration. The main requirement for the aerodynamic model was that it should reflect major nonlinear aerodynamic phenomena at high angles of attack. The SUPRA aerodynamic model was validated at high angles of attack using comparison with a dynamically scaled free-spinning model in a vertical wind tunnel. The aerodynamic model was successfully validated by a number of expert pilots and found acceptable for upset recovery training.

2.7 Key Conclusions

The following key conclusions were drawn from the literature survey:

- Flight envelope upset is characterised by high angular rates, angles of attack and sideslip angles that are outside the aerodynamic envelope, pitch angles and bank angles that are outside the attitude envelope, unusual climb rates or descent rates, airspeeds that are either too low or too high, and normal load factors that are outside the structural integrity envelope.
- The Airplane Upset Recovery Training Aid and the Airplane Flying handbook provide pilot procedures and techniques for manual flight envelope recovery, including stall recovery, pitch angle recovery, bank angle recovery, and spin recovery. The manual flight upset recovery procedures are rules-based with actions that are typically performed one action at a time, sequentially. The recovery procedures only cover a limited number of specific upset scenarios.
- The manual upset recovery techniques all prescribe that if the aircracft is stalled, then it must first be recovered from stall, before proceeding with attitude and airspeed recovery.
- Research on automatic flight envelope protection and recovery tends to focus more on flight envelope *protection* and less on flight envelope *recovery*. The research also tends to focus more on handling sensor failures, actuator failures and airframe damage, and less on recovering from out-of-envelope flight conditions.
- The majority of the research found on flight envelope extension and recovery is applied to high performance fighter aircraft rather than civil transport aircraft. Fighter aircraft are designed to operate over much wider flight envelopes than civil transport aircraft, and are also equipped with actuators that have much greater control authority. Civil transport aircraft are designed for more limited flight envelopes and do not have access to specialised actuators such as thrust vectoring, canards, or forebody strakes. The flight envelope recovery strategies designed for agile fighter aircraft are therefore not directly applicable to the more sluggish transport aircraft.
- The main control techniques employed for flight envelope extension and flight envelope recovery are gain-scheduled control, linear parameter-varying control, feedback linearisation, nonlinear dynamic inversion, and sliding control. In many cases, the design of the control gains are supported by a bifurcation analysis of the aircraft's nonlinear dynamics.
- Very little research has been performed on the development of automatic flight envelope recovery systems for large transport aircraft. The research has rather been focussed on improving ground-based flight simulators for large transport aircraft to more accurately simulate the aircraft behaviour in flight envelope upset conditions. The main objectives of the research is to extend the aerodynamic model for large transport aircraft into the stall and the post-stall regime, and to more accurately represent the motion cues provided to the pilots in the flight simulators.

2.8 Research Gaps and Expected Contributions

The following research gaps were identified in the literature survey, and it is expected that research contributions can be made in these areas:

- The majority of the research found on flight envelope recovery is applied to high performance fighter aircraft rather than large transport aircraft. A research contribution can

therefore be made by developing an automated envelope recovery strategy that is suitable for the dynamics and actuation capabilities of a large transport aircraft, rather than for a fighter aircraft.

- The flight envelope recovery techniques found in the literature do not exploit the time-scale separation between the fast rigid body rotational dynamics and the slow point mass translational dynamics of the aircraft. A research contribution can therefore be made by conceptually splitting the full flight envelope recovery into an "inner loop recovery" of the angular rates and aerodynamic envelope, and an "outer loop recovery" of the attitude, flight path angle, and airspeed. The angular rates and the aerodynamic envelope recovery would be performed through controlling the fast rotational dynamics, and the attitude, flight path angle and airspeed recovery would be performed through controlling the slower point mass translational dynamics.

- The flight envelope recovery techniques found in the literature all use active feedback control that is designed using explicit quantitative knowledge of the wide-envelope aerodynamic model. The techniques include fixed-gain control, gain-scheduled control, linear parameter varying control, adaptive control, feedback linearisation, and nonlinear dynamic inversion. A research contribution can therefore be made by designing a passive aerodynamic recovery method that does not require explicit quantitative knowledge of the wide-envelope aerodynamic model, but uses implicit qualitative knowledge of the natural stability of the aircraft to reduce high angular rates and to recover the aerodynamic envelope.

- Although the *manual* flight envelope recovery procedures are performed as a sequence of actions, few of the *automatic* flight envelope recovery techniques perform the flight envelope recovery in stages. The existing automatic flight envelope recovery techniques also do not exploit the flight envelope protection functions that are provided by the fly-by-wire control systems of modern large transport aircraft. A research contribution can therefore be made by developing a state-machine based attitude and flight vector recovery function that uses the conventional fly-by-wire flight control laws with their protection functions to recover the attitude, flight path angle and airspeed.

- No Lyapunov-based controllers were found in literature for flight envelope recovery. A research contribution can therefore be made by developing a Lyapunov-based inner-loop controller that actively recovers the angular rates and aerodynamic envelope of the aircraft.

- The automated flight envelope recovery techniques found in literature do not attempt to perform the attitude, flight path angle, and airspeed recovery in an optimal way. A research contribution can therefore be made by formulating the flight envelope recovery as an optimal control problem with the objective to find the optimal state trajectories and control inputs sequences that minimise the total altitude lost during the recovery maneuver.

2.9 Research Decisions

Based on the information gained from the literature survey, as well as the research gaps identified, the following research decisions were made:

- The flight envelope recovery will be defined to encompass high angular rate recovery, aerodynamic envelope recovery, attitude recovery, flight path angle recovery, and airspeed

recovery while respecting the load factor and maximum airspeed constraints of the structural integrity envelope.

- The NASA Generic Transport Model will be used as the basis for the analysis of the wide-envelope flight dynamics of large transport aircraft, and for the design and verification of the flight envelope recovery system. The Simulink model has been sourced from NASA Langley Research Center and permission has been obtained to use the model for this research.

- The time scale separation between the fast rigid body rotational dynamics and the slow point mass translational dynamics of the aircraft will be exploited to split the flight envelope recovery into an "inner loop recovery" of the angular rates and aerodynamic envelope, and an "outer loop recovery" of the attitude, flight path angle, and airspeed.

- A two-tier flight envelope recovery strategy is proposed that first performs high angular rate recovery and aerodynamic envelope recovery, and then performs attitude recovery, flight path angle recovery, and airspeed recovery. We argue that angular rate and aerodynamic envelope recovery should be performed first, since it is an enabler for attitude, flight path angle and airspeed vector recovery. This strategy agrees with the prescribed pilot upset recovery procedures for large transport aircraft.

- Two approaches to aerodynamic envelope recovery will be investigated: a passive aerodynamic envelope recovery that uses the natural stability of the aircraft to recover the aerodynamic envelope and does not require explicit knowledge of the forces and moments produced by the control surfaces, and an active aerodynamic envelope recovery that employs a Lyapunov-based inner-loop controller to actively recover the angular rates and aerodynamic envelope using the control surfaces of the aircraft.

- Two approaches to attitude, flight path angle and airspeed recovery will be investigated: a state machine based attitude and flight vector recovery function that uses the conventional fly-by-wire flight control laws with their protection functions, and an optimal attitude and flight vector recovery function that simultaneously recovers the bank angle, flight path angle, and airspeed of the aircraft from upset conditions while minimising the total altitude lost during the recovery maneuver.

- The time scale separation assumption between the aerodynamic envelope recovery and the attitude, flight path angle and airspeed recovery will be validated by implementing the integrated flight envelope recovery system in simulation and verifying its performance on the full NASA Generic Transport Model simulation.

Chapter 3

The NASA Generic Transport Model

This chapter provides more information on the NASA Generic Transport Model (GTM) that was chosen as the research platform for the automatic flight envelope recovery research. The GTM is a 5.5% dynamically scaled unmanned aerial vehicle (UAV) of a large transport aircraft that was developed by NASA for experimental flight tests outside the normal flight envelope. Wide-envelope aerodynamic models were developed for the GTM based on wind tunnel tests over an extensive range of angles of attack, sideslip angles, angular rates, and control surface deflections. A full nonlinear simulation model of the GTM was created and made available by NASA for passenger aircraft upset recovery research.

The chapter provides some background on NASA's development and validation of the GTM, presents the mathematical models of the aircraft dynamics and wide-envelope aerodynamics that are implemented by the NASA GTM, and gives a high-level description of the NASA GTM Simulink model provided for research use.

3.1 Development and Validation

The NASA Generic Transport Model, shown in figure 3.1, is a 5.5% dynamically scaled unmanned aerial vehicle (UAV) of a large transport aircraft that was developed by NASA for experimental flight tests outside the normal flight envelope.

Figure 3.1: The NASA Generic Transport Model (image supplied with Simulink model)

Starting in 1999, the NASA Aviation Safety Program initiated research that focused on investigating flight dynamics and control of large transport aircraft in upset conditions. As part of these efforts, NASA developed the Airborne Subscale Transport Aircraft Research (AirSTAR) flight test facility, which includes a 5.5% dynamically scaled unmanned aerial vehicle (UAV), called the Generic Transport Model (GTM), for experimental flight tests outside the normal

flight envelope that are considered too risky for full-scale manned transport airplane testing [47].

The aircraft is equipped with an inertial measurement system that provides three-axis linear acceleration measurements, three-axis angular rate measurements, estimated attitude angles, and GPS velocity and position. The aircraft is also equipped with anemometric sensors that measure the angle of attack, sideslip angle, dynamic pressure and static pressure. The measured static pressure and ambient temperature are used to determine the air density and altitude, and the dynamic pressure is used to calculate airspeed. The engine speed is also measured through the engine control unit [41].

Aerodynamic models were developed for the GTM based on wind tunnel tests conducted at the NASA Langley Research Center using a model that has the same mold line specifications [2, 40, 48, 49]. Wind tunnel tests were performed over an extensive range of angles of attack, sideslip angles, angular rates, and control surface deflections. The wind tunnel test program included both static and dynamic wind tunnel tests. In static wind tunnel tests, the aircraft model is kept stationary during the test to capture the contributions of angle of attack, sideslip angle, and control surface deflections on the aerodynamic forces and moments. In dynamic wind tunnel tests, the aircraft model is given spinning or sinusoidal motions during the test to capture the contributions of the aircraft angular rates on the aerodynamic forces and moments. The dynamic wind tunnel tests included both rotary balance testing and forced oscillation testing.

Rotary balance testing consists of rotating the aircraft model at constant angular rates about an axis parallel to the incoming air velocity vector. The aircraft therefore performs a coning motion about the airspeed velocity vector. The independent variables during the test are typically angle of attack, sideslip angle, and rotation rate, and may also include control surface deflections. The angle of attack and sideslip angle are normally held constant during the rotation. Rotary balance data is usually used to predict spin modes, since the rotary balance motion is representative of non-oscillatory steady spin motion.

Forced oscillation testing consists of moving the aircraft model in a sinusoidal manner independently about each of the roll, pitch, and yaw axes during the wind tunnel test. The independent variables during the test are the nominal angle of attack, and the amplitude and frequency of the oscillation. Additional independent variables may include sideslip angle and control surface deflections.

The rotary balance data and the forced oscillation data were blended using a Hybrid-Kalviste method to model the total contribution of the aircraft angular rates to the aerodynamic forces and moments acting on the aircraft [2].

The aerodynamic model was validated using data obtained from free-spin testing of a dynamically scaled model. Free-spin testing involves hand-launching a dynamically scaled model into a vertically rising airstream at a high angle of attack and rotation rate. The model will typically enter a stable spin mode if one or more exists for the given configuration and control surface deflections. The validation results showed that the aerodynamic model captured the aerodynamic characteristics of a large transport aircraft during a spin with sufficient accuracy to enable simulation of spins that are consistent with free-spin wind tunnel test data.

From these aerodynamic models, a full non-linear flight dynamics simulation model of the GTM was created and made available by NASA for research use.

3.2 Mathematical Model

This section gives an overview of the mathematical model of the NASA GTM flight dynamics. The axis systems and notation are established, the standard six degrees of freedom equations of motion and general forces and moments model for an aircraft are briefly presented, and finally the detailed implementation of the wide-envelope aerodynamic model for the NASA GTM is discussed.

3.2.1 Axis Systems and Notation

Three axis systems are used in the mathematical modelling of the NASA GTM flight dynamics, namely the inertial axis system, the body axis system, and the wind axis system. These axis systems are quite standard in aircraft flight dynamics modelling and will therefore be discussed only briefly to establish the conventions used.

Inertial Axis System

The standard North-East-Down (NED) axis system is used as the inertial reference frame. The NED axis system assumes a flat, non-rotating earth and adequately approximates an inertial axis system for the short flight distances considered in this project. The centre of the axis system is chosen to coincide with some convenient reference point on the earth's surface. The x-axis points in the North direction, the y-axis points in the East direction, and the z-axis completes the right-handed, orthogonal axis system and points in the Down direction.

Body Axis System

The body axis system is fixed to the aircraft body with its origin chosen to coincide with the aircraft's centre of mass. The x-axis lies in the plane of symmetry and points through the nose along the zero angle of attack line of the wing. The y-axis lies perpendicular to the plane of symmetry in the direction of the starboard wing. The z-axis completes the right-handed, orthogonal axis system and points downwards through the underside of the aircraft.

Wind Axis System

The wind axis system is similar to the body axis system in that its origin coincides with the aircraft's centre of mass and therefore moves with the aircraft. However, the x-axis points in the direction of the velocity vector, the z-axis lies in the aircraft's plane of symmetry pointing generally downwards through the underside of the aircraft, and the y-axis completes the right-handed, orthogonal axis system pointing generally in the direction of the starboard wing.

Notation

The following standard notation is used in the aircraft model

X, Y, Z:	Coordinates of the force vector in body axes (axial, lateral, and normal force)
L, M, N:	Coordinates of the moment vector in body axes (roll, pitch, and yaw moment)
U, V, W:	Coordinates of the linear velocity vector in body axes (axial, lateral, and normal velocity)
P, Q, R:	Coordinates of the angular velocity vector in body axes (roll, pitch, and yaw rates)
$\delta_A, \delta_R, \delta_E$:	Aileron, rudder, and elevator deflections. A positive deflection is defined as one that produces a negative moment.

Airspeed, Angle of Attack, and Sideslip Angle

The airspeed magnitude $\overline{V}$, the angle of attack α, and the sideslip angle β are important variables for calculating the aerodynamic forces and moments acting on the aircraft.

Informally, the angle of attack is the angle at which the incoming airflow strikes the aircraft from "below the wings", while the sideslip angle is the angle at which the incoming airflow strikes the aircraft from the "side". More formally, the angle of attack α is the angle between the aircraft body x-axis and the projection of the aircraft velocity vector into the body xz-plane,

while the sideslip angle β is the angle between the aircraft velocity vector and its own projection into the body xz-plane. The airspeed magnitude $\overline{V}$ is the magnitude of the aircraft velocity vector. Note that the airspeed magnitude, angle of attack, and sideslip angle must be calculated from the aircraft velocity relative to the air, and not its velocity relative to the ground.

The airspeed magnitude $\overline{V}$, the angle of attack α, and the sideslip angle β are related to the components of the aircraft airspeed velocity vector (U, V, W) in the body axes through the following relationship

$$\overline{V} = \sqrt{U^2 + V^2 + W^2} \tag{3.1}$$

$$\alpha = \arctan \frac{W}{U} \tag{3.2}$$

$$\beta = \arcsin \frac{V}{\overline{V}} \tag{3.3}$$

The inverse relationship is

$$U = \overline{V} \cos\alpha \cos\beta \tag{3.4}$$

$$V = \overline{V} \sin\beta \tag{3.5}$$

$$W = \overline{V} \sin\alpha \cos\beta \tag{3.6}$$

3.2.2 Six Degrees of Freedom Equation of Motion

Kinetics

The kinetic equations of motion relate the forces and moments acting on the aircraft to the kinematic state of the aircraft, i.e. its position, velocity and acceleration. The kinetic equations of motion in their classic form where all vectors are coordinated in body axes are given as

$$m\dot{\boldsymbol{V}} = \boldsymbol{F} - \boldsymbol{\omega} \times m\boldsymbol{V} \tag{3.7}$$

$$\mathbf{I_B}\dot{\boldsymbol{\omega}} = \boldsymbol{M} - \boldsymbol{\omega} \times \mathbf{I_B}\boldsymbol{\omega} \tag{3.8}$$

where $\boldsymbol{V}$ and $\boldsymbol{\omega}$ are the velocity and angular rate vectors of the aircraft, m and $\mathbf{I_B}$ are the aircraft mass and moment of inertia matrix, and $\boldsymbol{F}$ and $\boldsymbol{M}$ are the force and moment vectors acting on the aircraft, all coordinated in the body axes.

Equations 3.7 and 3.8 relate the forces and moments acting on the aircraft to the time rate of change of its linear velocity and angular rate. The cross product terms arise because the force, moment, velocity and angular rate vectors have been coordinated in body axes, and not in inertial axes. Given the forces and moments that act on the body, as well as its mass and moment of inertia, the linear velocity and angular rate of the aircraft can be propagated over time.

Kinematics

The kinematic equations of motion relate the motion variables, such as linear velocity, angular rate, position and attitude, to each other over time. The position and attitude of the aircraft are represented by the following variables

N, E, D: Coordinates of the position vector in inertial axes (north, east and down position)

Φ, Θ, Ψ: The Euler 3-2-1 attitude parameters of the body axis system relative to the inertial axis system (roll, pitch, and yaw angle)

Attitude Representation

The Euler attitude represents the attitude of the body axis system relative to the inertial axis system using three Euler angles (roll, pitch and yaw) and a predefined order of rotation. The Euler 3-2-1 sequence starts with the two axis systems aligned and then rotates the body axis system through the following set of ordered rotations

1. Yaw the body axis system by rotating it about its z-axis through the yaw angle Ψ
2. Pitch the resulting first intermediate axis system about its y-axis through the pitch angle Θ
3. Roll the resulting second intermediate axis system about its x-axis through the roll angle Φ

The coordinates of a vector in the inertial axis system may be transformed to coordinates in the body axis system using the direction cosine matrix (DCM), which may be written as a function of the Euler angles as follows

$$\begin{bmatrix} x_B \\ y_B \\ z_B \end{bmatrix} = \begin{bmatrix} C_\Psi C_\Theta & S_\Psi C_\Theta & -S_\Theta \\ C_\Psi S_\Theta S_\Phi - S_\Psi C_\Phi & S_\Psi S_\Theta S_\Phi + C_\Psi C_\Phi & C_\Theta S_\Phi \\ C_\Psi S_\Theta C_\Phi + S_\Psi S_\Phi & S_\Psi S_\Theta C_\Phi - C_\Psi S_\Phi & C_\Theta C_\Phi \end{bmatrix} \begin{bmatrix} x_I \\ y_I \\ z_I \end{bmatrix} \tag{3.9}$$

Conversely, the coordinates of a vector in the body axis system may be transformed to coordinates in the inertial axis system by using the inverse direction cosine matrix. Since the DCM is an orthogonal matrix, its matrix inverse is simply its transpose. Therefore

$$\begin{bmatrix} x_I \\ y_I \\ z_I \end{bmatrix} = \begin{bmatrix} C_\Psi C_\Theta & C_\Psi S_\Theta S_\Phi - S_\Psi C_\Phi & C_\Psi S_\Theta C_\Phi + S_\Psi S_\Phi \\ S_\Psi C_\Theta & S_\Psi S_\Theta S_\Phi + C_\Psi C_\Phi & S_\Psi S_\Theta C_\Phi - C_\Psi S_\Phi \\ -S_\Theta & C_\Theta S_\Phi & C_\Theta C_\Phi \end{bmatrix} \begin{bmatrix} x_B \\ y_B \\ z_B \end{bmatrix} \tag{3.10}$$

Attitude Kinematics

The time rate of change of aircraft attitude is related to the aircraft body angular rates through the following equation, which uses the Euler 3-2-1 parameterisation

$$\begin{bmatrix} \dot{\Phi} \\ \dot{\Theta} \\ \dot{\Psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin\Phi\tan\Theta & \cos\Phi\tan\Theta \\ 0 & \cos\Phi & -\sin\Phi \\ 0 & \sin\Phi\sec\Theta & \cos\Phi\sec\Theta \end{bmatrix} \begin{bmatrix} P \\ Q \\ R \end{bmatrix} \tag{3.11}$$

Position Kinematics

The time rate of change of the aircraft position is related to the aircraft velocity coordinated in body axes through the following equation

$$\begin{bmatrix} \dot{N} \\ \dot{E} \\ \dot{D} \end{bmatrix} = \begin{bmatrix} C_\Psi C_\Theta & S_\Psi C_\Theta & S_\Theta \\ S_\Psi C_\Theta & S_\Psi S_\Theta S_\Phi + C_\Psi C_\Phi & C_\Theta S_\Phi \\ -S_\Theta & S_\Psi S_\Theta C_\Phi - C_\Psi S_\Phi & C_\Theta C_\Phi \end{bmatrix} \begin{bmatrix} U \\ V \\ W \end{bmatrix} \tag{3.12}$$

3.2.3 Forces and Moments Model

The total force and moment acting on the aircraft may be written as the sum of aerodynamic, thrust and gravitational forces and momoments, as follows

$$\boldsymbol{F} = \boldsymbol{F^A} + \boldsymbol{F^E} + \boldsymbol{F^G} \tag{3.13}$$
$$\boldsymbol{M} = \boldsymbol{M^A} + \boldsymbol{M^E} + \boldsymbol{M^G} \tag{3.14}$$

where the superscripts A, E, and G denote aerodynamic, engine, and gravitational components respectively.

Aerodynamic Forces and Moments

The aerodynamic forces and moments are functions of the air density ρ, the airspeed magnitude $\overline{V}$, the angle of attack α, the sideslip angle β, the angular rates of the aircraft with respect to inertial space (P, Q, R), and the aerodynamic control surface deflections $\boldsymbol{\delta}$.

$$\boldsymbol{F^A} = \boldsymbol{f^A}(\alpha, \beta, P, Q, R, \boldsymbol{\delta}, \rho, \overline{V}) \tag{3.15}$$
$$\boldsymbol{M^A} = \boldsymbol{m^A}(\alpha, \beta, P, Q, R, \boldsymbol{\delta}, \rho, \overline{V}) \tag{3.16}$$

$\boldsymbol{F^A}$ and $\boldsymbol{M^A}$ are the aerodynamic force and moment vectors, and $\boldsymbol{f^A}$ and $\boldsymbol{m^A}$ are both general multivariable nonlinear functions that are determined by the aerodynamic characteristics of the specific aircraft. The wide-envelope aerodynamic model for the NASA GTM will be discussed in more detail in section 3.2.4.

Engine Forces and Moments

The engine forces and moments are produced by the engines of the aircraft, and are functions of the throttle settings of the engines $\boldsymbol{\delta_T}$, as well as the air density and the airspeed.

$$\boldsymbol{F^E} = \boldsymbol{f^E}(\boldsymbol{\delta_T}, \rho, \overline{V}) \tag{3.17}$$
$$\boldsymbol{M^E} = \boldsymbol{m^E}(\boldsymbol{\delta_T}, \rho, \overline{V}) \tag{3.18}$$

$\boldsymbol{F^E}$ and $\boldsymbol{M^E}$ are the engine force and moment vectors, and $\boldsymbol{f^E}$ and $\boldsymbol{m^E}$ are general multivariable nonlinear functions that are determined by the characteristics of the specific engines used on the aircraft.

The engine force vector lies primarily along the positive body x-axis, but there may also be small components in the body y-axis and body z-axis directions due to the alignment of the engines relative to the aircraft body. The engines on typical commercial airliners are rotated slightly upwards and inwards towards the fuselage. Assuming that the engines are set to produce equal thrust, the y-axis thrust components should cancel out, but the z-axis components will not, which means that a small component of thrust may be expected in the body z-axis.

The engine moment vector is the result of the engine thrust vectors not acting through the aircraft centre of mass, as well as the gyroscopic torques associated with the angular momentums of the two engines. On aircraft with underwing-mounted engines, the dominant effect of the engines is to produce a nose-up pitching moment that is proportional to the total engine thrust, due to the thrust vector acting through a point below the centre of mass. The rolling and yawing moments due to the thrust vectors not acting through the centre of mass tend to oppose each other and cancel out, due to the symmetry of the aircraft and the fact that the left and right engines are normally operated to produce equal thrust. The angular momentums of the engines also tend to oppose each other and cancel out, due to the fact that the engines are designed to rotate in opposite directions and are typically operated at equal engine speeds.

Gravitational Forces and Moments

The gravitational forces and moments acting on the aircraft are modelled using a simple gravitational model that assumes a constant gravitational acceleration that does not vary with latitude or altitude. The standard value for gravitational acceleration is used, and it is assumed to point downwards in the inertial axis system. The gravitational forces and moments coordinated into body axes are functions of the attitude of the body axis system relative to the inertial axis system, and the mass of the aircraft.

$$\boldsymbol{F^G} = \boldsymbol{f^G}(\Phi, \Theta) \tag{3.19}$$
$$\boldsymbol{M^G} = \boldsymbol{0} \tag{3.20}$$

$\boldsymbol{F^G}$ and $\boldsymbol{M^G}$ are the gravitational force and moment vectors coordinated into the body axis system, while Φ and Θ are the roll and pitch angles of the aircraft. (The gravitational force is independent of the yaw angle Ψ of the aircraft.) Since the gravitational force acts through the centre of mass of the aircraft, and since the gravity gradient moment acting on the aircraft is negligible, the gravitational moments are assumed to be zero.

3.2.4 Wide-Envelope Aerodynamic Model

This section describes the wide-envelope aerodynamic model of the NASA GTM. The wide-envelope aerodynamic model implements the general multivariable nonlinear functions represented by equations 3.15 and 3.16 in section 3.2.3.

The aerodynamic forces and moments are modelled by the following set of standard equations

$$X^A = \frac{1}{2}\rho\overline{V}^2 S C_X \tag{3.21}$$
$$Y^A = \frac{1}{2}\rho\overline{V}^2 S C_Y \tag{3.22}$$
$$Z^A = \frac{1}{2}\rho\overline{V}^2 S C_Z \tag{3.23}$$
$$L^A = \frac{1}{2}\rho\overline{V}^2 S b C_l \tag{3.24}$$
$$M^A = \frac{1}{2}\rho\overline{V}^2 S \bar{c} C_m \tag{3.25}$$
$$N^A = \frac{1}{2}\rho\overline{V}^2 S b C_n \tag{3.26}$$
$$\tag{3.27}$$

where (X^A, Y^A, Z^A) and (L^A, M^A, N^A) are the aerodynamic forces and moments coordinated in the body axes, ρ is the air density, $\overline{V}$ is the airspeed magnitude, S is the wing area, b is the wing span, $\bar{c}$ is the mean aerodynamic chord, and C_X, C_Y, C_Z, C_l, C_m, and C_n are non-dimensional aerodynamic coefficients. The non-dimensional aerodynamic coefficients capture the aerodynamic properties of the shape of the aircraft, independently from of its size. The wing area, wing span, and mean aerodynamic chord parameters are used to scale the aerodynamic forces and moments according to the size of the aircraft. The aerodynamic forces and moments are also proportional to the quantity $\frac{1}{2}\rho\overline{V}^2$, which is called the dynamic pressure.

The non-dimensional aerodynamic coefficients of the NASA GTM are modelled by the following set of equations

$$
\begin{aligned}
C_X &= C_{X,static}(\alpha,\beta) + \Delta C_{X,\delta}(\alpha,\beta,\delta_E,\delta_A,\delta_R) \\
&\quad + \Delta C_{X,\hat{q}_{osc}}(\alpha,\hat{q}_{osc}) + \Delta C_{X,\hat{\omega}_{ss}}(\alpha,\beta,\hat{\omega}_{ss}) && (3.28)\\
C_Y &= C_{Y,static}(\alpha,\beta) + \Delta C_{Y,\delta}(\alpha,\beta,\delta_A,\delta_R) \\
&\quad + \Delta C_{Y,\hat{p}_{osc}}(\alpha,\hat{p}_{osc}) + \Delta C_{Y,\hat{r}_{osc}}(\alpha,\hat{r}_{osc}) + \Delta C_{Y,\hat{\omega}_{ss}}(\alpha,\beta,\hat{\omega}_{ss}) && (3.29)\\
C_Z &= C_{Z,static}(\alpha,\beta) + \Delta C_{Z,\delta}(\alpha,\beta,\delta_E,\delta_A,\delta_R) \\
&\quad + \Delta C_{Z,\hat{q}_{osc}}(\alpha,\hat{q}_{osc}) + \Delta C_{Z,\hat{\omega}_{ss}}(\alpha,\beta,\hat{\omega}_{ss}) && (3.30)\\
C_l &= C_{l,static}(\alpha,\beta) + \Delta C_{l,\delta}(\alpha,\beta,\delta_A,\delta_R) \\
&\quad + \Delta C_{l,\hat{p}_{osc}}(\alpha,\hat{p}_{osc}) + \Delta C_{l,\hat{r}_{osc}}(\alpha,\hat{r}_{osc}) + \Delta C_{l,\hat{\omega}_{ss}}(\alpha,\beta,\hat{\omega}_{ss}) && (3.31)\\
C_m &= C_{m,static}(\alpha,\beta) + \Delta C_{m,\delta}(\alpha,\beta,\delta_E,\delta_A,\delta_R) \\
&\quad + \Delta C_{m,\hat{q}_{osc}}(\alpha,\hat{q}_{osc}) + \Delta C_{m,\hat{\omega}_{ss}}(\alpha,\beta,\hat{\omega}_{ss}) && (3.32)\\
C_n &= C_{n,static}(\alpha,\beta) + \Delta C_{n,\delta}(\alpha,\beta,\delta_A,\delta_R) \\
&\quad + \Delta C_{n,\hat{p}_{osc}}(\alpha,\hat{p}_{osc}) + \Delta C_{n,\hat{r}_{osc}}(\alpha,\hat{r}_{osc}) + \Delta C_{n,\hat{\omega}_{ss}}(\alpha,\beta,\hat{\omega}_{ss}) && (3.33)
\end{aligned}
$$

The total aerodynamic coefficients consist of the sum of baseline static coefficients and incremental dynamic coefficients and incremental control surface coefficients. The baseline static coefficients C_{static} represent the effect of the angle of attack α and sideslip angle β only. The incremental dynamic coefficients represent the contributions of the angular rates (at the given angle of attack and sideslip angle). The incremental dynamic coefficients are split into those that were derived from the rotary balance data $\Delta C_{\hat{\omega}_{ss}}$ and those that were derived from the forced oscillation data $\Delta C_{\hat{p}_{osc}}$, $\Delta C_{\hat{q}_{osc}}$, and $\Delta C_{\hat{r}_{osc}}$. The incremental control surface coefficients ΔC_δ represent the contributions of the control surface deflections δ_A, δ_E, and δ_R (also at the given angle of attack and sideslip angle).

To understand how the incremental dynamic coefficients are implemented, we must first explain how the rotary balance data and the forced oscillation data are used in the aerodynamic model. The rotary balance data and the forced oscillation data are blended using a Hybrid-Kalviste method to model the total contribution of the aircraft angular rates to the aerodynamic forces and moments acting on the aircraft. The blending approach is to decompose the total angular rate vector $\bar{\Omega}$ into a "steady-state" component ω_{ss} along the aircraft velocity vector (corresponding to the rotary balance motion) and "oscillatory" components (Ω_{osc}) along the aircraft body axes (corresponding to the forced oscillation motion) as shown in figure 3.2.

Three cases are used when decomposing the total angular rate vector into the steady-state and oscillatory components, and they are shown in figure 3.3. In case 1, the projection of the total angular rate vector into the x-z plane is closer to the body z-axis, and it is decomposed into a steady-state component ω_{ss} and an oscillatory yaw rate component r_{osc}. In case 2, the projection of the total angular rate vector into the x-z plane is closer to the body x-axis, and it is decomposed into a steady-state component ω_{ss} and an oscillatory roll rate component p_{osc}. In case 3, the angular rate vector is uncoordinated (the roll rate and the yaw rate have opposite signs), and it is decomposed into oscillatory components only, with no steady-state component. The equations that are used to calculate steady-state and oscillatory rate components ω_{ss}, p_{osc}, q_{osc}, and r_{osc} from the body angular rate components p_b, q_b, and r_b are shown in table 3.1.

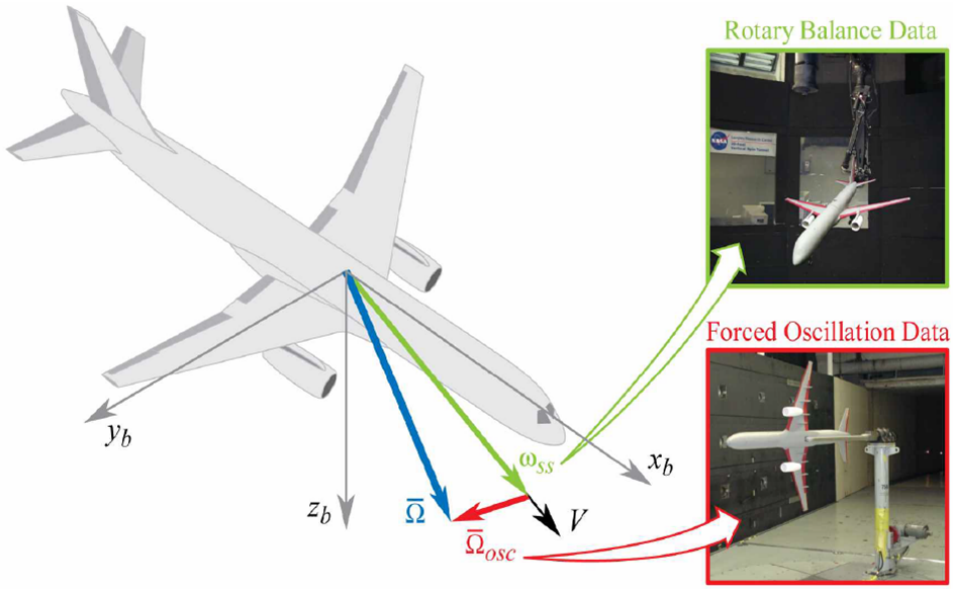


Figure 3.2: Decomposition of the total angular rate vector into steady-state and oscillatory components [2].

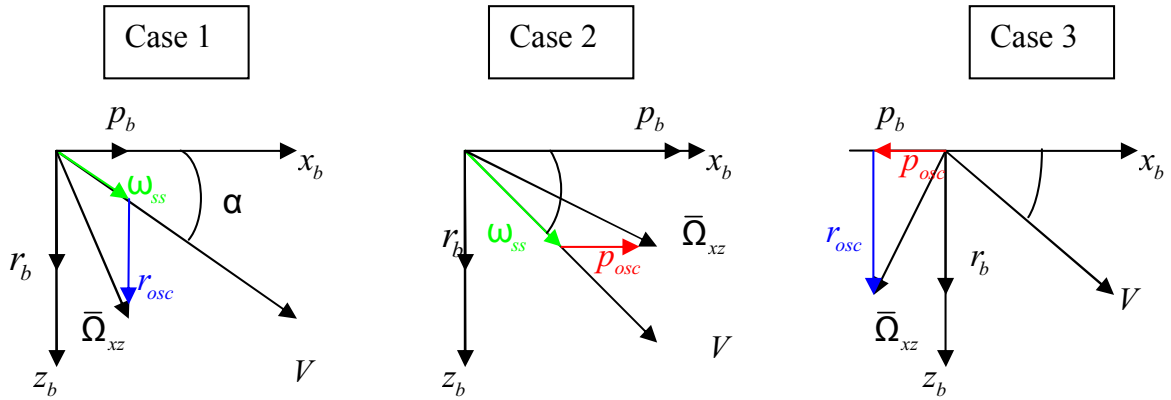


Figure 3.3: Three decomposition schemes used with the Kalviste methods [2].

Table 3.1: Equations used in the Hybrid Kalviste method [2].

Angular RateTerm	Case 1	Case 2	Case 3
$\omega_{ss} =$	$\frac{p_b}{\cos\alpha \cos\beta}$	$\frac{r_b}{\sin\alpha \cos\beta}$	0
$p_{osc} =$	0	$p_b - \omega_{ss} \cos\alpha \cos\beta$	p_b
$q_{osc} =$	$q_b - \omega_{ss} \sin\beta$	$q_b - \omega_{ss} \sin\beta$	q_b
$r_{osc} =$	$r_b - \omega_{ss} \sin\alpha \cos\beta$	0	r_b

Before using the steady-state component and oscillatory components in the wide-envelope aerodynamic model, they are first non-dimensionalised as follows

$$\hat{\omega}_{ss} = \frac{\omega_{ss} b}{2\overline{V}} \tag{3.34}$$

$$\hat{p}_{osc} = \frac{p_{osc} b}{2\overline{V}} \tag{3.35}$$

$$\hat{q}_{osc} = \frac{q_{osc} \bar{c}}{2\overline{V}} \tag{3.36}$$

$$\hat{r}_{osc} = \frac{r_{osc} b}{2\overline{V}} \tag{3.37}$$

The functions for each aerodynamic coefficient term in the wide-envelope aerodynamic model (Equations 3.28 to 3.33) are implemented as an n-dimensional lookup table, where n is equal to the number of independent variables for that term. Linear interpolation is used to calculate the coefficient term for values of the independent variables that are between table breakpoints, while linear extrapolation is used for values outside of the table breakpoints.

3.3 Simulation Model for Matlab Simulink

A Simulink model that is representative of the actual NASA GTM unmanned aerial vehicle is provided by NASA for research use, and is also used to perform simulation tests of proposed flight control algorithms before they are cleared for flight testing on the actual NASA GTM. A screenshot of the NASA GTM Simulink model (top level) is shown in figure 3.4.

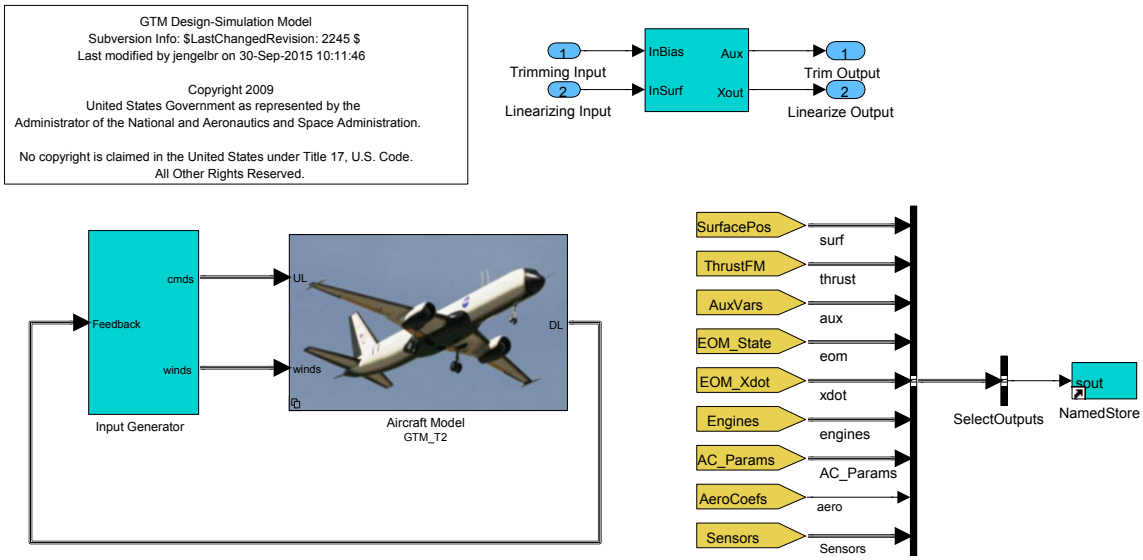


Figure 3.4: Simulink model of the NASA GTM (top level).

The Aircraft Model block contains the wide-envelope aircraft dynamics model as provided by NASA, and the Input Generator block is where proposed flight control algorithms must be implemented. The simulation model includes detailed models of the aircraft dynamics, the wide-envelope aerodynamics, the engines, the onboard sensors (including sensor bandwidth and measurement noise), and the control surface actuators (including actuator dynamics, saturation limits, and slew rate limits).

The aerodynamic forces and moments are modelled with nonlinear aerodynamic coefficients functions implemented as multi-dimensional lookup tables with contributions from static forces and moments due to angle of attack and sideslip angle, dynamic forces and moments due to the body angular rates, and control surface forces and moments due to the deflections of the ailerons, elevator, and rudder.

The engine models include a detailed model of the engine dynamics, the engine alignment, the throttle to engine RPM curve, the engine RPM to thrust curve, and the effect of atmospheric density on the engine thrust. The model outputs the engine thrust, engine speed, engine angular momentum, fuel flow rate, and exhaust gas temperatures. The engine forces and moments are calculated taking the centre of gravity location into account as well as the gyroscopic torques associated with the angular momentums of the engines.

The sensor models include detailed models of the onboard air data sensors, the inertial sensors, and GPS sensor. The air data sensors provide measurements of the angle of attack, sideslip angle, dynamic pressure, true airspeed, and altitude of the aircraft. The NASA GTM uses an onboard MIDG II combined inertial navigation system and GPS unit. The inertial navigation system provides measurements of the body angular rates, the specific accelerations, the attitude (bank angle, pitch angle, yaw angle), the aircraft position (latitude, longitude, altitude), and the inertial velocity (north, east, up). The sensor models include realistic sensor noise, as well as sensor biases, scale factors, and measurement range limits.

The actuator models include detailed models of the actuators for the ailerons, elevators, rudders, left and right engine throttles, spoilers, flaps, landing gear, nose wheel steering, and brakes. The actuator models include realistic actuator bandwidths, slew rate limits, range limits, dead band and time delays.

The aircraft dynamics takes into account changes in the mass and moment of inertia, and shifts in the centre of gravity location, as a function of fuel consumption, damage cases, and the state of the landing gear.

A Note on Dynamic Scaling

When considering simulation results obtained using the NASA GTM, one should bear in mind that the results are for a 5.5% dynamically scaled aircraft. The time scales of the aircraft dynamics and the flight envelope recovery may therefore seem unrealistically short for a large transport aircraft, the translational velocities may seem unrealistically low, and the angular rates may seem unrealistically high. To obtain a more realistic picture of the results for a full size large transport aircraft, the simulation results could be scaled dynamically, using the scale factors shown in table 3.2 [50].

Table 3.2: Dynamic scaling factors.

Scale factor		NASA GTM ($n = 1/0.055$)
Linear dimension	n	18.18
Linear velocity	$\sqrt{n}$	4.264
Linear acceleration	1	1
Angular displacement	1	1
Angular velocity	$1/\sqrt{n}$	0.2345
Time	$\sqrt{n}$	4.264

Chapter 4

Passive Aerodynamic Envelope Recovery

This chapter presents a passive approach to aerodynamic envelope recovery that uses the natural angular rate damping of the aircraft to recover from high angular rates, and uses the natural stability of the aircraft to recover the angle of attack and sideslip angle. An overview of the approach is provided, followed by some background theory on bifurcation analysis and continuation methods. A bifurcation analysis of the wide-envelope nonlinear simulation model for the Generic Transport Model (GTM) is performed with respect to elevator to find the desirable stable equilibria of the aircraft inside the aerodynamic envelope, and to check for undesirable stable equilibria outside the envelope. Nonlinear simulations are then performed to confirm the results of the bifurcation analysis and to analyse the transient behaviour during the stall and spin departure and recovery. The regions of attraction of the stable equilibria, both inside and outside the aerodynamic envelope, are explored by generating state trajectories of the fast rotational dynamics and by performing Monte Carlo simulations. The state trajectories provide a qualitative picture of the "flow" of the system and visually reveal the regions of attraction, but are only useful for visualising system dynamics that manifest dominantly in only two dimensions. The Monte Carlo simulations are used to explore higher-dimensional regions of attraction where the state trajectories manifest in three or more of the aircraft's state variables. The bifurcation analysis and nonlinear simulatons presented in this chapter have been published by the author in Engelbrecht et al [51] and Pauck and Engelbrecht [52].

4.1 Overview

As argued in the introduction and confirmed by the literature survey, the recovery from high angular rates and from adverse aerodynamics is the key first step of a full flight envelope recovery procedure. Once the angular rates and aerodynamic envelope have been recovered, the conventional flight control laws and protection functions are available again, and can be used for the next stages of attitude recovery, flight path angle recovery, and airspeed recovery. Recovery from high angular rates and adverse aerodynamics may be complicated by nonlinear behaviour and/or model uncertainties in the forces and moments produced by the control surfaces in the post-stall region. Also, the anemometric sensors that measure airspeed, angle of attack, and sideslip angle may intermittently provide invalid sensor measurements, for example if the angle of attack and/or sideslip angle fall outside the valid sensor ranges, and may therefore not be available for feedback control. It would be advantageous if the natural stability of the aircraft could be leveraged to reduce high angular rates and to recover the aerodynamic envelope, since this approach would not require explicit knowledge of the forces and moments produced by the control surfaces, and would not rely on wide-envelope sensor measurements from the

anemometric sensors.

A passive aerodynamic envelope recovery approach is therefore proposed that uses the natural angular rate damping of the aircraft to recover from high angular rates, and uses the natural tendency of the aircraft to point its nose into the airflow to recover the angle of attack and sideslip angle. The hypothesis is that the aircraft will be attracted to a stable equilibrium with low angular rates and a desirable angle of attack and sideslip angle inside the aerodynamic envelope. However, the danger is that the aircraft may have multiple stable equilibria, and may also naturally be attracted to undesirable angles of attack and sideslip angles outside the aerodynamic envelope, depending on the initial conditions. To determine the viability of the passive aerodynamic envelope recovery approach, all equilibrium states inside and outside the aerodynamic envelope should be determined, and their regions of attraction should be explored to indicate which initial conditions will stabilise to desirable equilibria inside the aerodynamic envelope, and which will stabilise to undesirable equilibria outside the aerodynamic envelope.

A bifurcation analysis of the wide-envelope nonlinear simulation model for the Generic Transport Model (GTM) is performed with respect to elevator to find the desirable stable equilibria of the aircraft inside the aerodynamic envelope, and to check for undesirable stable equilibria outside the envelope. For control surface deflections near zero, only one stable equilibrium branch is found, corresponding to the aircraft gliding with the steady-state angle of attack and sideslip angle well within the envelope. For excessive elevator deflections, two stable branches are found corresponding to stall and spin conditions. Nonlinear simulations are then performed to confirm the results of the bifurcation analysis and to analyse the transient behaviour during the stall and spin departure and recovery. The results confirm that the aircraft enters stall and spin when excessive elevator is commanded, and naturally recovers from stall and spin when the elevator is returned to zero deflection.

The regions of attraction of the stable equilibria, both inside and outside the aerodynamic envelope, are then explored by generating state trajectories of the fast rotational dynamics and by performing Monte Carlo simulations. The state trajectories provide a qualitative picture of the "flow" of the system and visually reveal the regions of attraction, illustrating the natural angular rate recovery and the natural angle of attack and sideslip angle recovery. The Monte Carlo simulations are used to explore higher-dimensional regions of attraction where the state trajectories manifest in three or more of the aircraft's state variables. The region of attraction of the stable equilibrium inside the aerodynamic envelope, produced by zero elevator deflection, is found to be the entire range of angular rates and angles of attack over which the Generic Transport Model is valid. Similarly, the regions of attraction for the two stable equiliria outside the aerodynamic envelope, produced by excessive elevator deflections, are also found to be the entire range of angular rates and angles of attack over which the model is valid.

It should be noted that the bifurcation analysis of the Generic Transport model at Stellenbosch University was performed in parallel with a research team from Bristol University. Each team performed the bifurcation analysis on their own, but information was exchanged and the results were compared. The team from Stellenbosch University only investigated the stationary equilibrium branches, while the team from Bristol University also investigated the periodic equilibrium branches. Both teams published their results in parallel [51–53].

It should also be noted that the bifurcation diagrams of the NASA GTM in this chapter were generated by another PhD student, Mr Simon Pauck, who was co-supervised by the author. Mr Pauck modified the NASA GTM Simulink model to be compatible with Matlab Dynamical Systems toolbox and generated the bifurcation diagrams under the supervision of the author. However, the analysis and interpretation of the bifurcation diagrams, the nonlinear time history simulations, and the region of attraction analyses through state trajectories and Monte Carlo simulations were performed by the author himself.

4.2 Bifurcation Theory

This section provides a brief overview of the bifurcation theory required to appreciate the bifurcation analysis performed on the wide-envelope aircraft dynamics of the NASA GTM. The textbook on *Nonlinear Dynamics and Chaos* by Strogatz [54] and the survey article by Paranjape et al [55] are the primary sources of the information.

The application of bifurcation theory and continuation methods to flight dynamics problems was introduced by Carroll and Mehra [56] and Zagaynov and Goman [57]. A report by Cummings [58] gives a concise description of the use of continuation methods for the analysis of aircraft flight dynamics. Paranjape et al [55] performed a thorough survey of the state-of-the-art of the use of bifurcation and continuation methods for the analysis of aircraft trim and stability.

The primary goal of the bifurcation analysis is to construct the equilibrium manifolds of the nonlinear system dynamics and to analyse the stability of the equilibrium states. The equilibrium manifold is constructed using a mathematical technique known as the continuation method. The continuation algorithm calculates an entire family of equilibrium solutions as a chosen parameter is varied. The parameter to be varied may be a system parameter, such as the aircraft mass or centre of gravity location, or may be a control input, such as elevator deflection or thrust setting. Once the equilibrium states have been traced out, the nonlinear aircraft model is linearised about each equilibrium state, and the local stability of each equilibrium is determined from its linearised dynamics.

Modern continuation and bifurcation software, such as AUTO2000 [59] can efficiently automate the entire process, and can also compute branches of limit cycle solutions with varying control parameter. Using bifurcation analysis along with nonlinear time history simulation can be very effective in studying stability and control problems in nonlinear flight dynamics.

4.2.1 Standard Bifurcation Analysis

A nonlinear dynamical system, such as an aircraft, can be modelled with a set of ordinary differential equations in the form

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{p}) \tag{4.1}$$

where $\mathbf{x}$ is the state vector, $\mathbf{u}$ is the control input vector, and $\mathbf{p}$ is a vector of system parameters. For the purposes of a bifurcation analysis, the differential equations in 4.1 can be rewritten in the following form

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \lambda) \tag{4.2}$$

where λ is a scalar parameter of interest that will be varied, chosen as one of the input variables in $\mathbf{u}$ or one of the system parameters in $\mathbf{p}$. It is assumed that all the other input variables and system parameters in $\mathbf{u}$ and $\mathbf{p}$ will be held fixed during the analysis. The parameter λ that is varied is often called the *continuation parameter*.

Bifurcation and continuation software typically performs two primary tasks:

1. The equilibrium states $\mathbf{x}^*$ of the system are determined as the continuation parameter λ is varied. This is achieved by a continuation algorithm that solves the set of nonlinear algebraic equations for the case where the time derivative of the state vector equals zero.

$$\mathbf{0} = \mathbf{f}(\mathbf{x}^*, \lambda) \tag{4.3}$$

The equilibrium states determined by the continuation algorithms may be either stationary solutions or periodic orbits, such as limit cycles.

2. The local stability of the equilibrium states are determined by numerically calculating the Jacobian matrix at each equilibrium state,

$$J(\mathbf{x}^*) = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}\Big|_{\mathbf{x}=\mathbf{x}^*} \tag{4.4}$$

The stability of the stationary solutions are determined from the eigenvalues of the Jacobian matrix, while the stability of the periodic orbits are determined from its Floquet multipliers.

The results of the bifurcation analysis are presented in the form of *bifurcation diagrams*. Bifurcation diagrams are plots of the equilibrium state $\mathbf{x}^*$ with respect to the continuation parameter λ. The bifurcation diagrams present the higher-dimensional equilibrium state as a set of two-dimensional projections with each diagram plotting one of the state variables with respect to the continuation parameter. The bifurcation diagrams reveal branches of stationary equilibria as well as brances of period equilibria such as limit cycles. The stability of each equilibrium branch is indicated by using full lines for stable equilibria and dashed lines for unstable equilibria. Points where the stability of the equilibrium changes are called bifurcation points and are also marked on the bifurcation diagram. The type of bifurcation at each bifurcation point is identified, based on the way that the stability changes, and follow-on actions may be taken, if desired.

4.2.2 Types of Bifurcation Points

The following types of bifurcation points are typically encountered in aircraft flight dynamics:

1. A *limit point* bifurcation is a bifurcation point where two equilibrium branches coming from the same side of the diagram, one stable and the other unstable, meet and annihilate each other. A limit point bifurcation is also called a *fold bifurcation* or a *turning point bifurcation* due to the fact that visually it looks like an equilibrium branch is turning back on itself. A limit point bifurcation is also, more technically, called a *saddle node bifurcation*. The name is derived from higher dimension cases where equilibrium points known as saddle points and nodes collide and annihilate.

2. A *transcritical bifurcation* is a bifurcation point where two equilibrium branches coming from the same side of the diagram, one stable and the other unstable, meet and exchange stability.

3. A *supercritical pitchfork bifurcation* is a bifurcation point where a stable equilibrium branch splits into three equilibrium branches, two stable branches and one unstable branch.

4. A *subcritical pitchfork bifurcation* is a bifurcation point where three equilibrium branches, one stable branch and two unstable branches, meet and become on unstable equilibrium branch.

5. A *Hopf bifurcation* is a bifurcation point where limit cycles are created. The Hopf bifurcation is called supercritical if stable limit cycles are created, and is called subcritical if unstable limit cycles are created.

4.3 Bifurcation Analysis of Wide-Envelope Aircraft Dynamics

A bifurcation analysis of the wide-envelope nonlinear simulation model for the Generic Transport Model (GTM) was performed with respect to elevator to find the desirable stable equilibria of the aircraft inside the aerodynamic envelope, and to check for undesirable stable equilibria outside the envelope. The purpose of the bifurcation analysis was only to identify the stable equilibria of the NASA GTM for passive aerodynamic envelope recovery, and not to perform a detailed investigation of all stationary and period solutions of the NASA GTM nonlinear dynamics. For a detailed investigation and discussion of all stationary and period solutions of the NASA GTM nonlinear dynamics, the reader is referred to the comprehensive study performed by Gill et al [53]

4.3.1 Conditioning the NASA GTM for the Bifurcation Analysis

The bifurcation analysis was performed on the GTM simulation model using the Matlab Dynamical Systems Toolbox. The simulation model was conditioned in several ways to make it compatible with the bifurcation software. To provide smooth equations for the continuation algorithm, the linear interpolation of lookup tables was changed to cubic spline interpolation. To reduce the computational load, components of the simulation model that are not required for the analysis of the natural dynamics, such as sensor and actuator bandwidths, saturation limits, slew rates, quantization blocks and delays, were removed, the lookup tables for the aerodynamic data were restructured to remove unnecessary interpolation calculations, and split-surface actuators were combined into single actuators.

4.3.2 Bifurcation Diagrams

For the bifurcation analysis, the aileron and rudder control surfaces were held fixed at their trim deflections, and only the elevator deflection was varied as the continuation parameter. The throttle was set to zero since the stall and spin recovery procedure in the FAA Aircraft Flying Handbook [7] recommends that the throttle be reduced to idle as the first step of the recovery. The bifurcation analysis was performed at an altitude of 800 feet and all physical aircraft parameters, such as mass, moment of inertia, and centre of gravity location, were set to the default values for the NASA GTM version 9.12.

The resulting bifurcation diagrams for elevator deflections from -30 to +5 degrees are shown in figures 4.1 to 4.8. The solid blue lines represent stable equilibrium branches and the dashed red lines represent unstable equilibrium branches. The black dots represent limit point bifurcation points and the purple stars represent Hopf bifurcation points. The bifurcation diagrams for the lateral state variables (sideslip angle, roll rate, yaw rate, and bank angle) are nearly symmetrical about the horizontal axis, due to the lateral symmetry of the aircraft. However, the diagrams are not perfectly symmetrical, due to a slight asymmetry in the wide-envelope aerodynamic model constructed from the wind tunnel data.

For $\delta_e = +5°$ to -2.4°, there is a single stable equilibrium branch with an angle of attack increasing from 0 to 10 degrees, and the sideslip angle and body angular rates remaining at zero. For δ_e = -2.4° to δ_e = -30°, the single stable branch splits into three branches: a stable branch with a positive roll rate, an unstable branch with a zero roll rate, and another stable branch with a negative roll rate. As the elevator deflection is varied from δ_e = -2.4° to δ_e = -30°, the angle of attack of the two stable branches increases from 10 to 21 degrees, and the angle of attack of the unstable branch increases from 10 to 23 degrees. Four unconnected unstable equilibrium branches are also observed, two unstable branches with high positive roll rates, and two unstable branches with high negative roll rates, spanning the entire range of elevator deflection. These unstable branches are not of great concern, since the aircraft state will not be attracted to them, and their stability does not change as a function of elevator deflection.

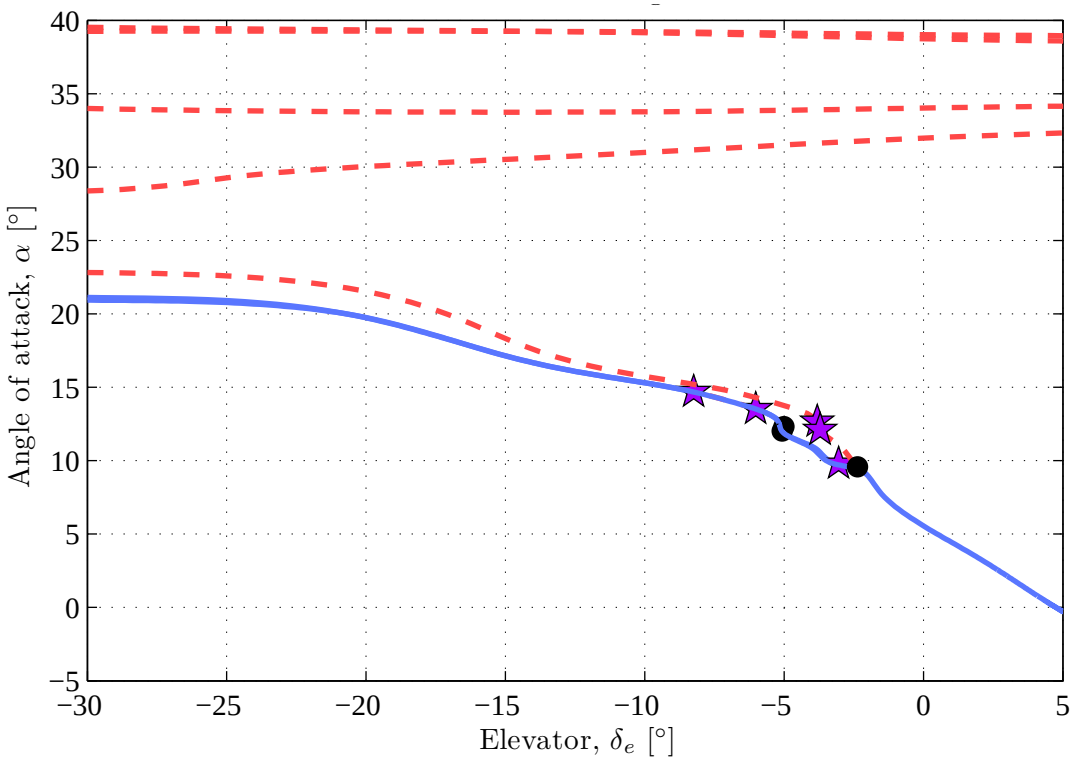


Figure 4.1: Bifurcation diagram: angle of attack vs. elevator deflection

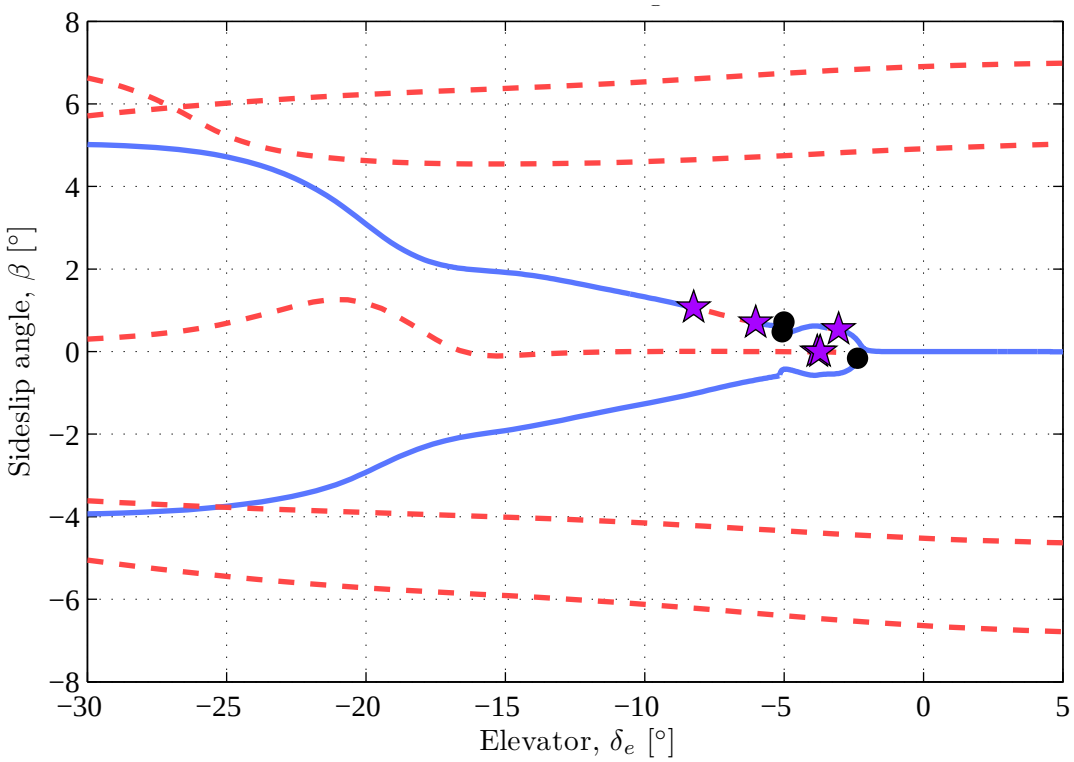


Figure 4.2: Bifurcation diagram: sideslip angle vs. elevator deflection

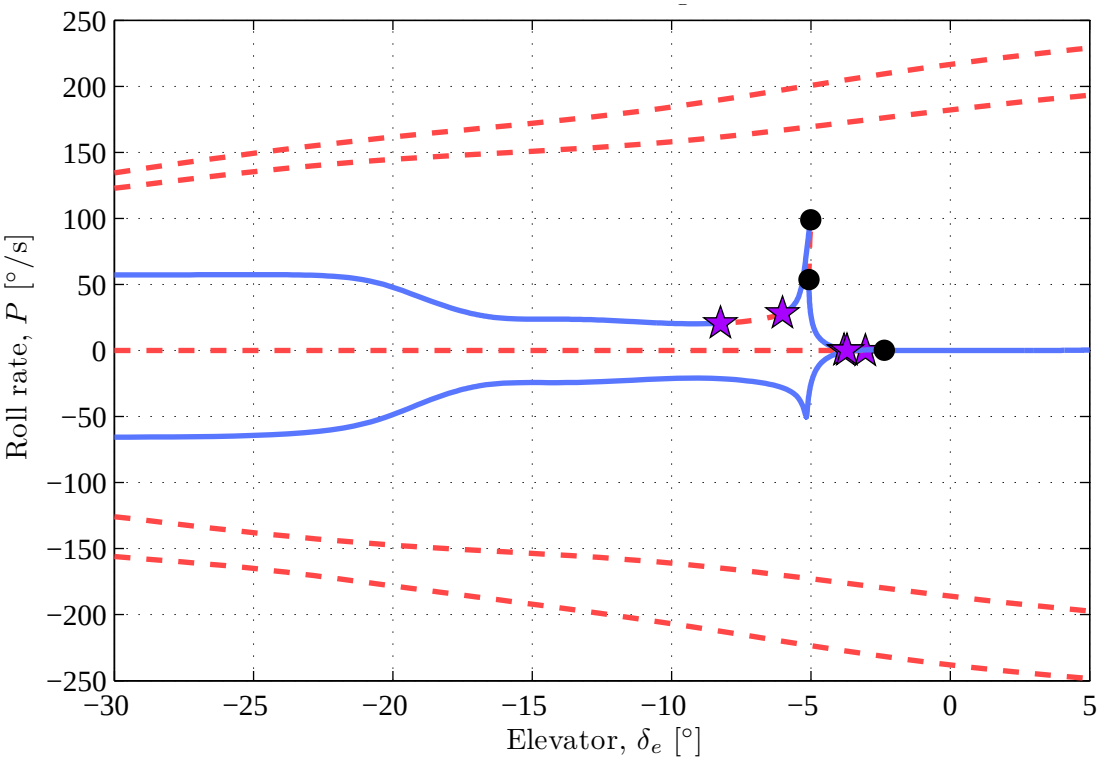


Figure 4.3: Bifurcation diagram: roll rate vs. elevator deflection

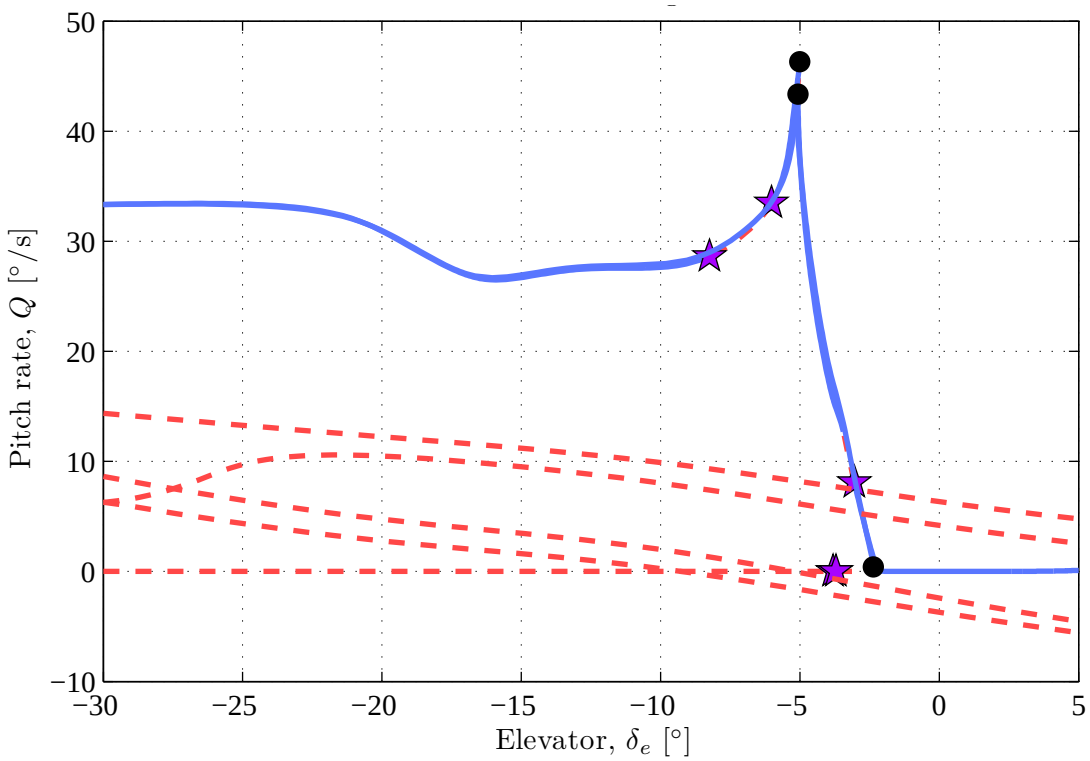


Figure 4.4: Bifurcation diagram: pitch rate vs. elevator deflection

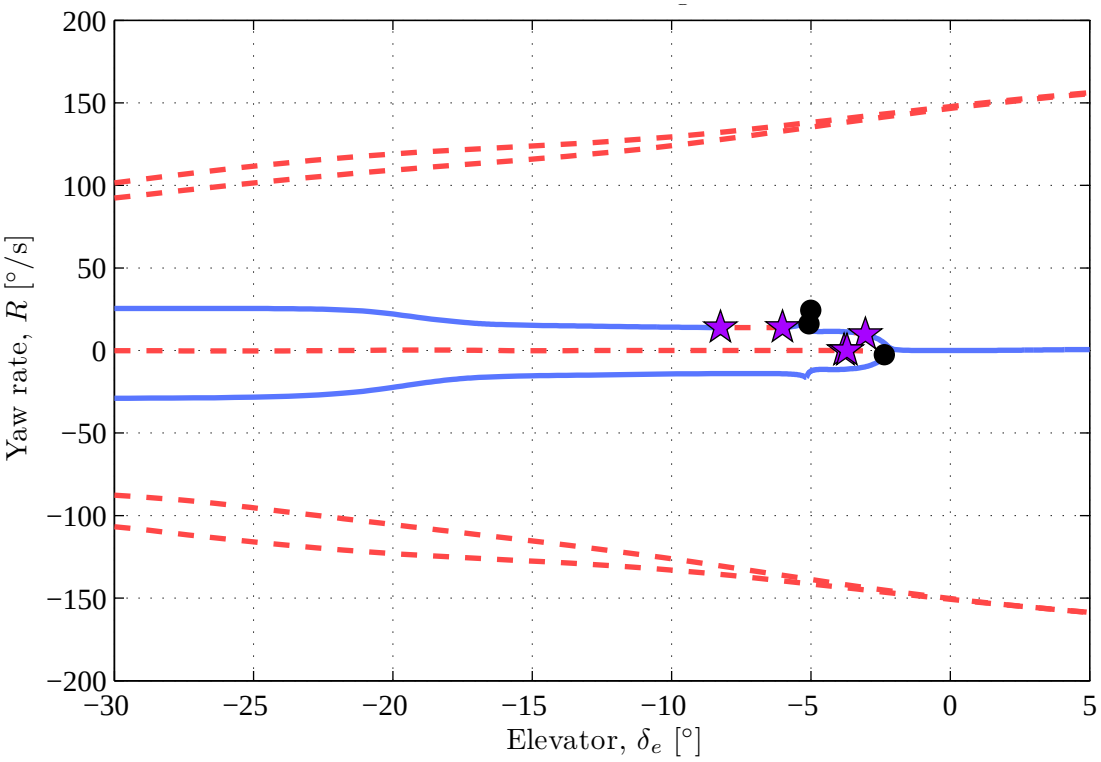


Figure 4.5: Bifurcation diagram: yaw rate vs. elevator deflection

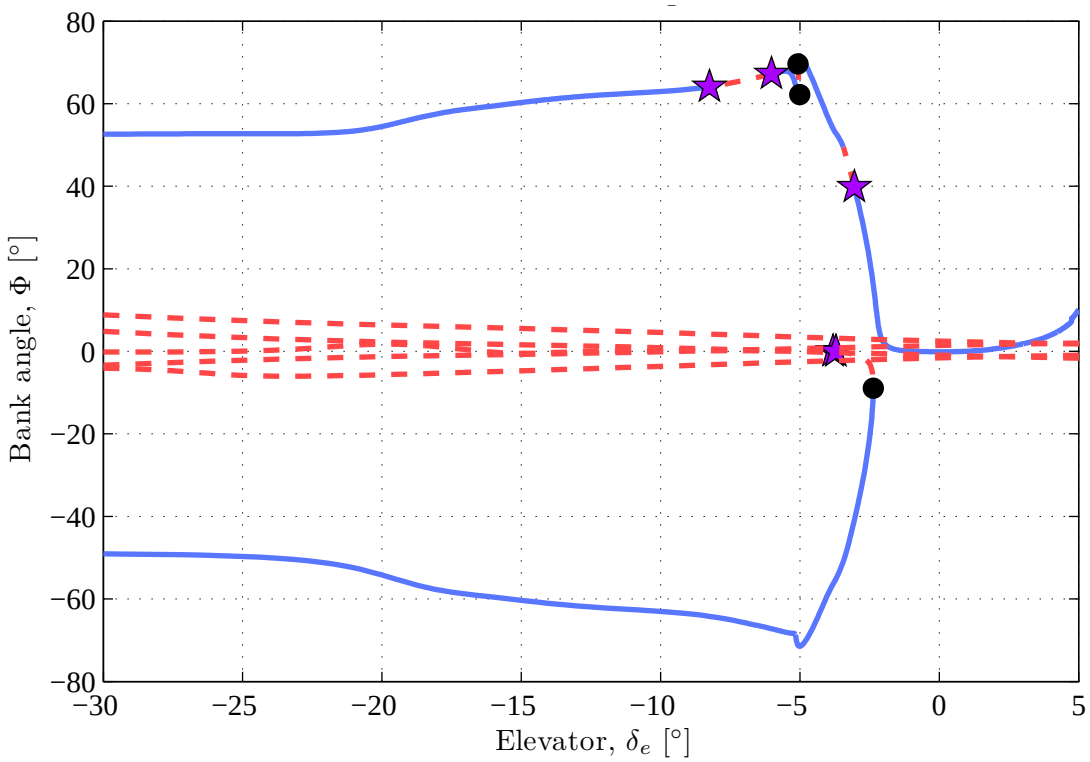


Figure 4.6: Bifurcation diagram: bank angle vs. elevator deflection

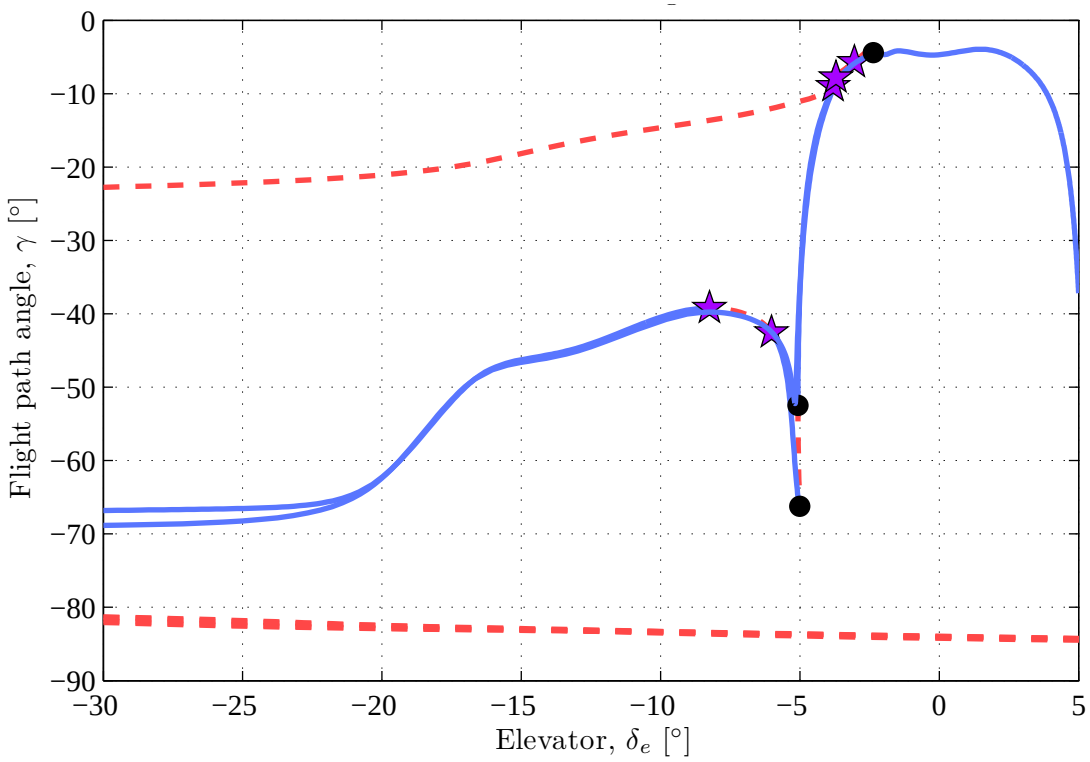


Figure 4.7: Bifurcation diagram: flight path angle vs. elevator deflection

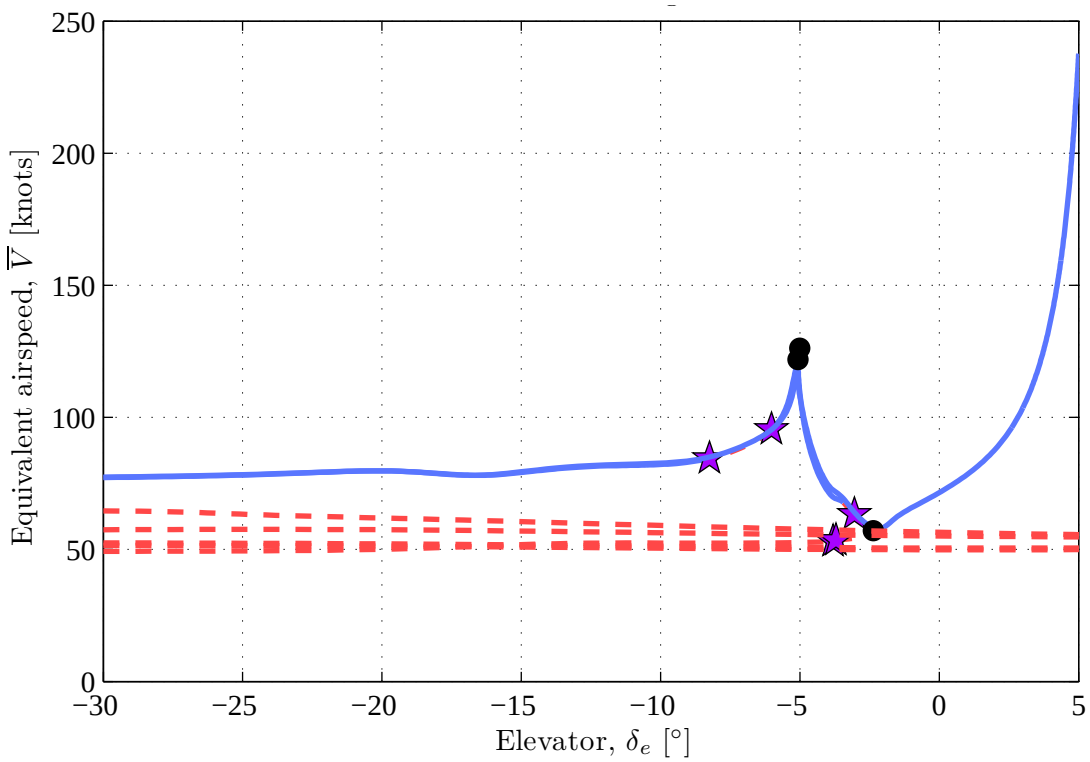


Figure 4.8: Bifurcation diagram: equivalent airspeed vs. elevator deflection

The following bifurcation points are observed in the bifurcation diagrams: a limit point bifurcation at δ_e = -2.4°, Hopf bifurcations between δ_e = -3° and δ_e = -3.8°, two limit point bifurcations at δ_e = -5°, and Hopf bifurcations between δ_e = -6° and δ_e = -8.2°. The point where the single stable branch splits into two stable branches and one unstable branch is actually not a supercritical pitchfork bifurcation, because the stable positive roll rate branch and the unstable zero roll rate branch are disconnected from the stable negative roll rate branch. The stable negative roll rate branch is the continuation of the single stable branch, while the unstable zero roll rate branch and the stable positive roll rate branch meet at a limit point bifurcation at δ_e = -2.4°. The two limit point bifurcation points at $\delta_e = -5°$ represent a roll rate hysteresis jump from 53 to 90 degrees per second as the elevator deflection moves through $\delta_e = -5°$ in the negative direction, and a roll rate hysteresis jump from 100 to 26 degrees per second as the elevator deflection moves through $\delta_e = -5°$ in the positive direction.

The stable limit cycles originating from the Hopf bifurcations were investigated in more detail by Gill et al [53]. The Hopf bifurcations create limit cycles corresponding to the phugoid mode becoming unstable and coupling into the lateral-directional dynamics. However, the frequency of these phugoid mode limit cycles are very low, with a period of approximately 30 seconds, and are therefore not considered as flight envelope upsets.

4.3.3 Interpretation of Bifurcation Analysis Results

The bifurcation diagrams show that for elevator deflections near zero, only one desirable stable equilibrium branch is found inside the aerodynamic envelope, and that no undesirable stable equilibria are found outside the envelope. Although it is difficult to prove that no other stable equilibria exist, an extensive search for disconnected branches was performed using simulation, trimming functions and continuation along secondary parameters, and none were found. The desirable stable equilibrium corresponds to the aircraft gliding under the influence of gravity with an equilibrium angle of attack and sideslip angle inside the aerodynamic envelope (figures 4.1 and 4.2), body angular rates near zero (figures 4.3, 4.4, and 4.5), the bank angle near wings level (figure 4.6) and a shallow, descending flight path angle (figure 4.7). For example, if the elevator deflection were set to 0 degrees, the aircraft would stabilise at an angle of attack of about 5 degrees, a sideslip angle of 0 degrees, body angular rates of 0 deg/s, a bank angle of 0 degrees, and a flight path angle of about -5 degrees. The presence of this desirable stable attractor, and the absence of stable undesirable attractors, indicate that the aircraft will naturally damp its body angular rates and recover its aerodynamic envelope, as long as the throttle is set to idle and all control surfaces are set to their neutral deflection angles.

For large elevator deflections, the bifurcation diagrams show that the single stable equilibrium branch splits into two stable branches and one unstable branch. The two stable branches correspond to two stable post-stall spins in opposite directions. The aircraft stalls, rolls to an unusually high bank angle (as shown in figure 4.6) and enters a stable post-stall spin with high body angular rates (as shown in figures 4.3, 4.4 and 4.5) in a downward spiral with a steep descending flight path angle (as shown in figure 4.7). In the post-stall spin, the aircraft exits the aerodynamic envelope and exhibits a high angle of attack, and a small sideslip angle (as shown in figures 4.1 and 4.2). For example, if the elevators were deflected to -30 degrees, the aircraft would stall to a high angle of attack of 21 degrees, roll to an unusual bank angle of about 50 degrees, and enter a stable spin in a steep downward spiral with a flight path angle of about -70 degrees. The spin motion would be observed in the body angular rates, with a roll rate of 58 deg/s, a pitch rate of 33 deg/s, and a yaw rate of 25 deg/s. The angle of sideslip would remain small (less than 5 degrees). The presence of these two undesirable stable attractors, and the fact that the desirable equilibrium is unstable, indicates that maintaining the elevator at large deflections will prevent the aircraft from naturally damping the body angular rates and from naturally recovering the aerodynamic envelope.

The important conclusions for upset recovery are that the natural stability of the aircraft will eventually recover it from stall and/or spin if the throttle is set to zero and all the control surfaces are returned to their neutral deflections, but that maintaining large nose-up elevator deflections will prevent the aircraft from recovering. However, the bifurcation analysis provides no information on the speed of the recovery. The transient behaviour of the recovery therefore needs to be analysed with nonlinear simulations.

4.4 Example Spin Entry and Passive Recovery

Simulations were performed to confirm the results of the bifurcation analysis and to analyse the transient behaviour during recovery of the aerodynamic envelope.

4.4.1 Simulation Setup

The simulations were performed with the GTM simulation model with the aircraft trimmed for straight and level flight at an equivalent airspeed of 75 knots and an altitude of 8000 feet. At 10 seconds into the simulation, the throttle was set to idle to remove power effects and to allow the aircraft to glide under the influence of gravity. At 20 seconds into the simulation, the elevator was deflected to -30 degrees to induce stall and post-stall spin. At 60 seconds into the simulation, the elevator was returned to zero deflection to allow natural recovery from stall and spin.

4.4.2 Simulation Results

The simulated time histories are shown in figures 4.9 to 4.12. The aircraft maintains normal flight during the first 10 seconds. After the throttle is cut, the aircraft starts gliding with a slow phugoid motion about a shallow flight path angle of about -5 degrees. When the elevator is set to -30 degrees deflection at t = 20 seconds, the aircraft stalls to an angle of attack of 21 degrees and a sideslip angle of 4 degrees. The aircraft rolls to an average bank angle of 50 degrees and enters a stable spin in a steep downward spiral with a flight path angle of about -68 degrees. The spin is observed in the body angular rates with a roll rate of 55 deg/s, a pitch rate of 34 deg/s, and a yaw rate of 25 deg/s. All the equilibrium values observed in the nonlinear simulation therefore correspond well with the values predicted by the bifurcation analysis.

When the elevators are returned to their zero deflections at 60 seconds, the aircraft naturally stops spinning and recovers its aerodynamic envelope. Within about 2 seconds, the body angular rates are damped to near zero, the angle of attack is recovered to 5 degrees and the sideslip angle is recovered to 0 degrees. Once the aerodynamic envelope is recovered, the aircraft performs a phugoid-like motion, repeatedly exchanging airspeed and altitude. Secondary angle of attack excursions are observed as a result of the phugoid-like motion. However, once the aerodynamic envelope is recovered, the conventional flight control laws are available again and can be used to perform attitude and flight path angle recovery and prevent the phugoid-like motion from occurring. Interestingly, the bank angle also naturally tends to recover to wings level, due to the stable wing dihedral and the fact that the aircraft accelerates towards the ground due to gravity.

The trajectory of the aircraft is shown in figure 4.13 and clearly shows the initial straight and level flight, followed by the steep downward spiral of the post-stall spin, and finally the passive stall and spin recovery.

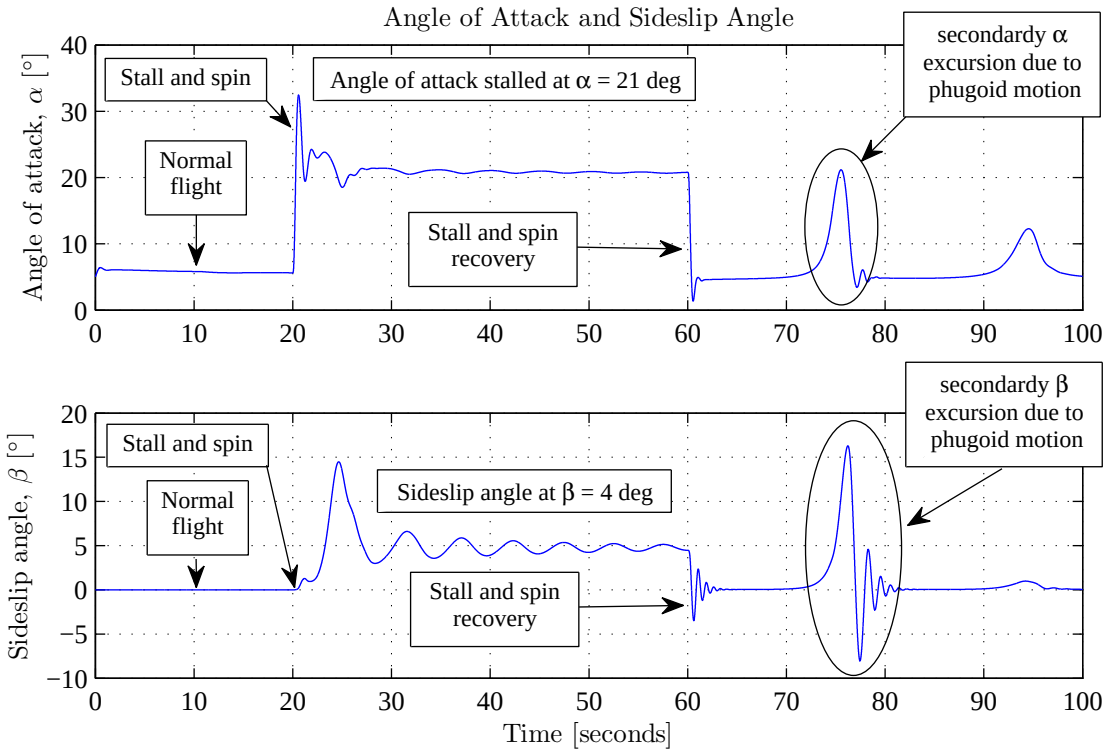


Figure 4.9: Stall and spin entry and passive recovery: angle of attack and sideslip angle vs. time

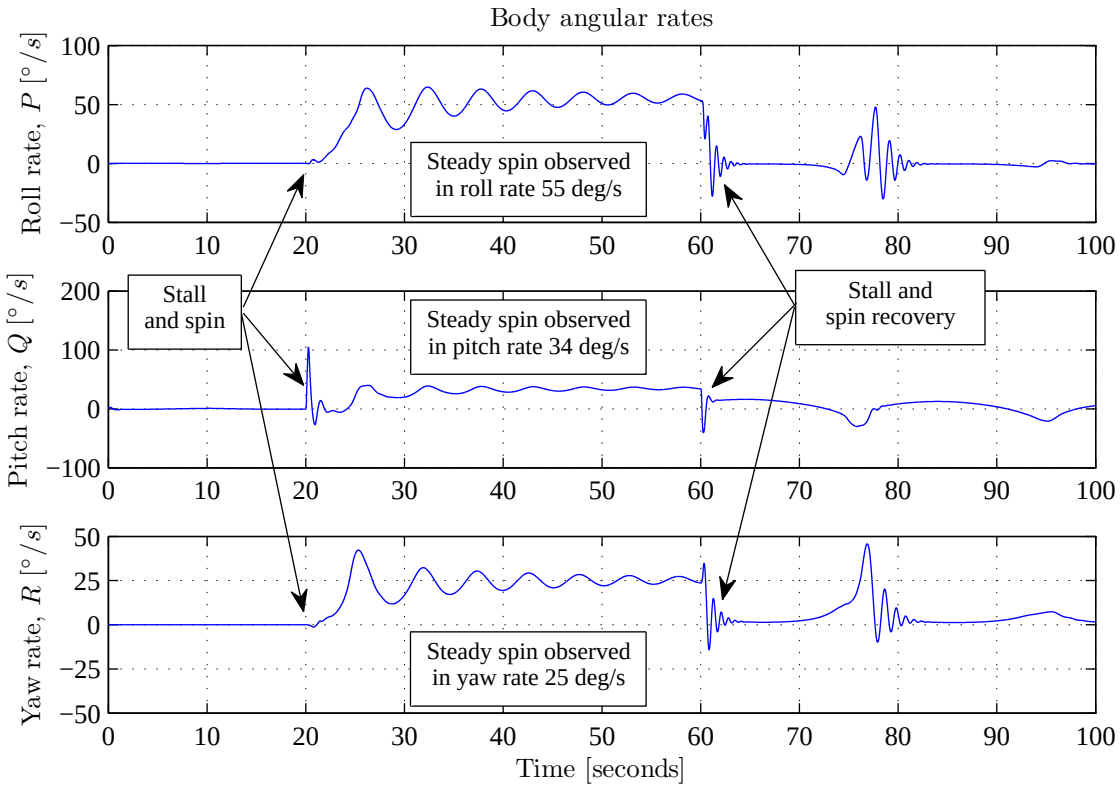


Figure 4.10: Stall and spin entry and passive recovery: body angular rates vs. time

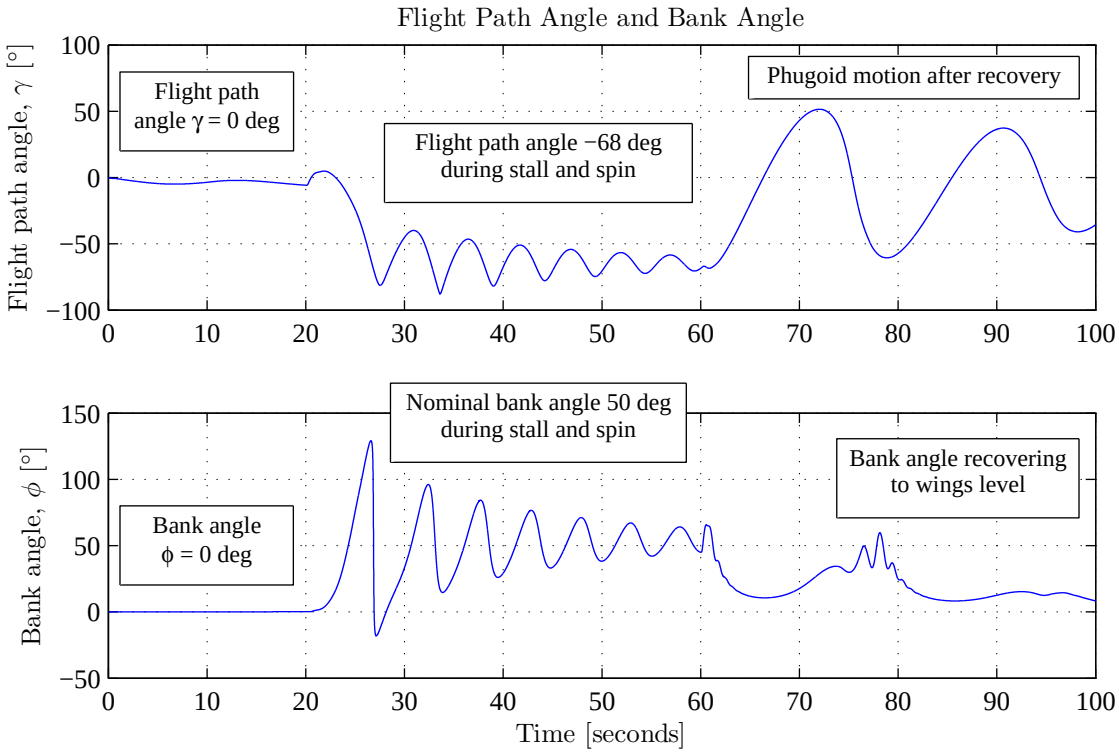


Figure 4.11: Stall and spin entry and passive recovery: flight path angle and bank angle vs. time

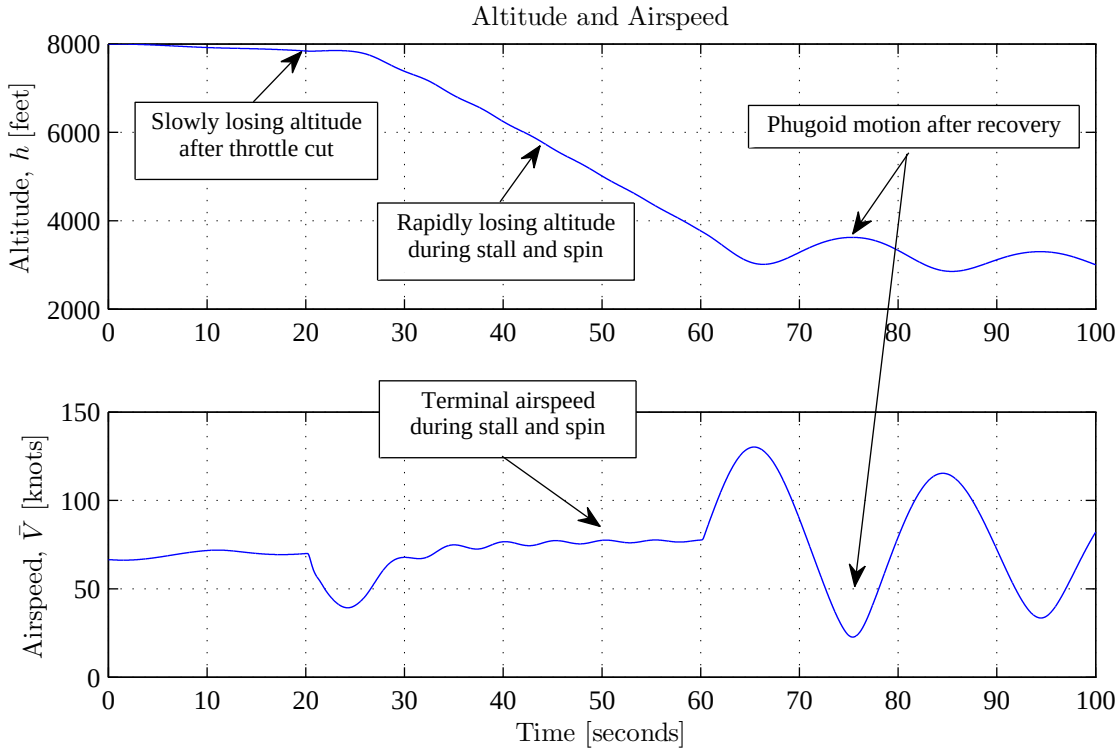


Figure 4.12: Stall and spin entry and passive recovery: altitude and airspeed vs. time

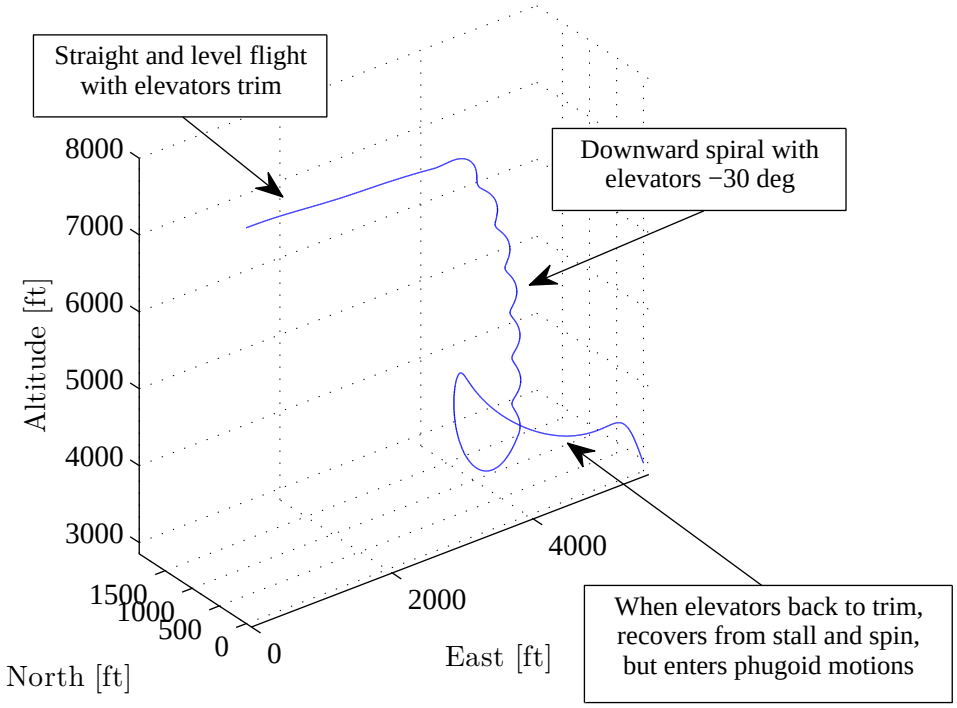


Figure 4.13: Stall and spin entry and passive recovery: three-dimensional flight trajectory in NED coordinates.

4.5 Regions of Attraction

Although the bifurcation analysis reveals stable equilibria towards which the aircraft state may be attracted, it does not provide information about the regions of attraction around the equilibria. In this section, the regions of attraction of the stable equilibria, both inside and outside the aerodynamic envelope, are explored by generating state trajectories of the fast rotational dynamics and by performing Monte Carlo simulations. The state trajectories are generated by simulating and plotting the state trajectories of the system from a grid of initial states. The state trajectorys provide a qualitative picture of the "flow" of the system and visually reveal the regions of attraction. However, the state trajectories are only useful for visualising system dynamics that manifest dominantly in only two dimensions.

Monte Carlo simulations are used to explore higher-dimensional regions of attraction where the state trajectories manifest in three or more of the aircraft's state variables. The Monte Carlo simulations are performed by simulating and plotting the time histories from a comprehensive number of initial states uniformly distributed over the set of valid states for the Generic Transport Model. The Monte Carlo simulations are not very useful for visualising "flow" of the system trajectories, but rather provide a useful statistical perspective on the regions of attraction. From the Monte Carlo results, the percentage of trajectories that are attracted to each equilibrium are determined, as well as the number of trajectories that tend towards infinity. The statistical distribution of the transient response settling time is also analysed. Scatter plots are used to reveal which initial states are attracted to which stable equilibria, by mapping each initial state to its corresponding final equilibrium state.

4.5.1 Note on Sufficient Number of Monte Carlo Simulations

When performing Monte Carlo simulations, the question is often asked how many Monte Carlo simulations are enough. In this section, Monte Carlo simulations are used to explore the regions of attraction of given stable equilibria. A certain region will be postulated to be a region of attraction of a given stable equilibrium, and then Monte Carlo simulations will be performed with initial states uniformly sampled from this region. If the simulated trajectory converges to the stable equilibrium, then the initial state is in its region of attraction. If the simulated trajectory does not converge to the stable equilibrium, then the initial state is not in its region of attraction. The Monte Carlo simulations can therefore be considered experiments with one of two possible outcomes, namely "in the region of attraction" and "not in the region of attraction", and the distribution of the outcomes is therefore a binomial distribution.

The rule of three in statistics states that if the outcomes of an experiment is binomially distributed and a given outcome does not occur within n trails, then one can say with 95% confidence that the probability of the outcome occurring is approximately $3/n$ or less. This means that if a set of n Monte Carlo simulations are performed, and all of the trajectories converge to the stable equilibrium, then the probability of a trajectory *not* converging to the stable equilibrium is $3/n$ or less, with a 95% confidence. For the region of attaction experiments in this section, sets of 1000 Monte Carlo simulations were performed, which implies that if all 1000 trajectories converge to the stable equilibrium, the probability of a trajectory not converging is less then 0.003 with a 95% confidence.

4.5.2 Passive Angle of Attack Recovery

The region of attraction for passive angle of attack recovery was explored by simulating the state trajectories from a grid of initial angles of attack from -5 to +85 degrees, with a zero initial sideslip angle, and with zero initial angular rates. The elevator deflection was held at a constant zero degrees, and the engine throttle was held fixed at idle. The state trajectories of angle of attack versus pitch rate are shown in figure 4.14 and the time histories are shown in figure 4.15.

The state trajectories show that all trajectories converge to the stable equilibrium inside the aerodynamic envelope and that no trajectories diverge to infinity. For all initial angles of attack from -5 to +85 degrees, the angle of attack returns to the stable equilibrium at $\alpha = 5.5$ degrees. A statistical analysis of the time histories shows that the angle of attack recovery has an average 5% settling time of 0.8 seconds, with a standard deviation of 0.3 seconds.

The region of attraction for passive angle of attack recovery therefore spans the range of angles of attack from -5 to +85 degrees, and the angle of attack is passively recovered within about 0.8 seconds.

It should be noted that for some trajectories with very high initial angles of attack, the angle of attack of the trajectory goes below -5 degrees, which is outside the lookup table range of the NASA GTM wide-envelope aerodynamic model. For these trajectories the lookup tables are extrapolated calculate the aerodynamic forces and moments.

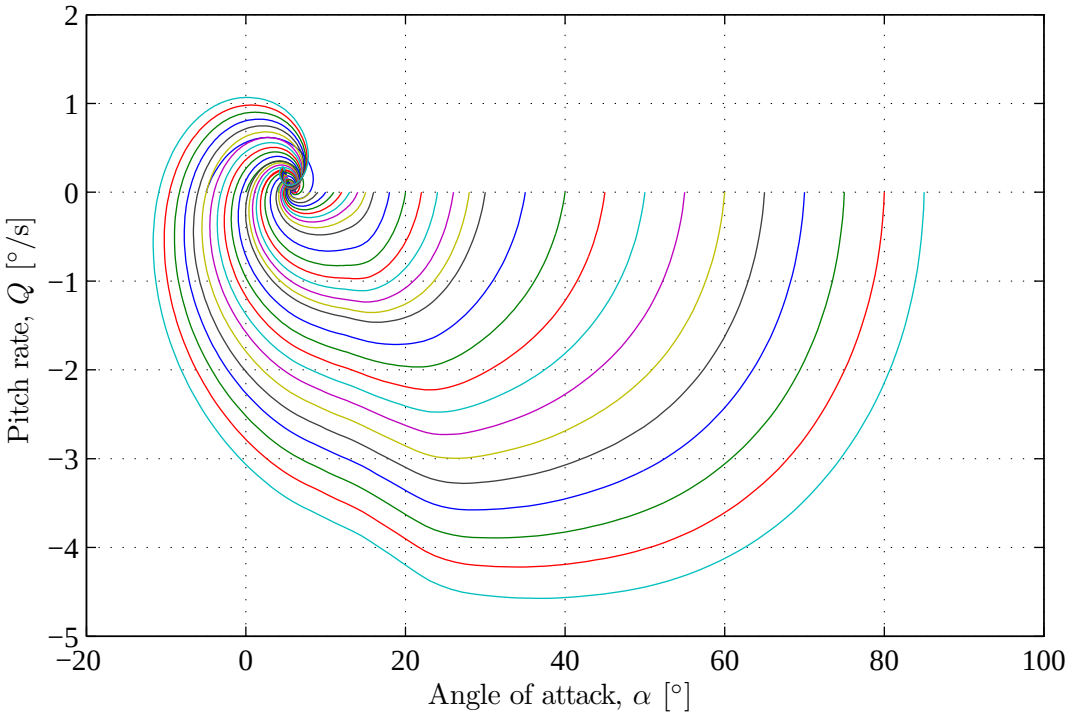


Figure 4.14: Passive angle of attack recovery: state trajectories of angle of attack vs. pitch rate

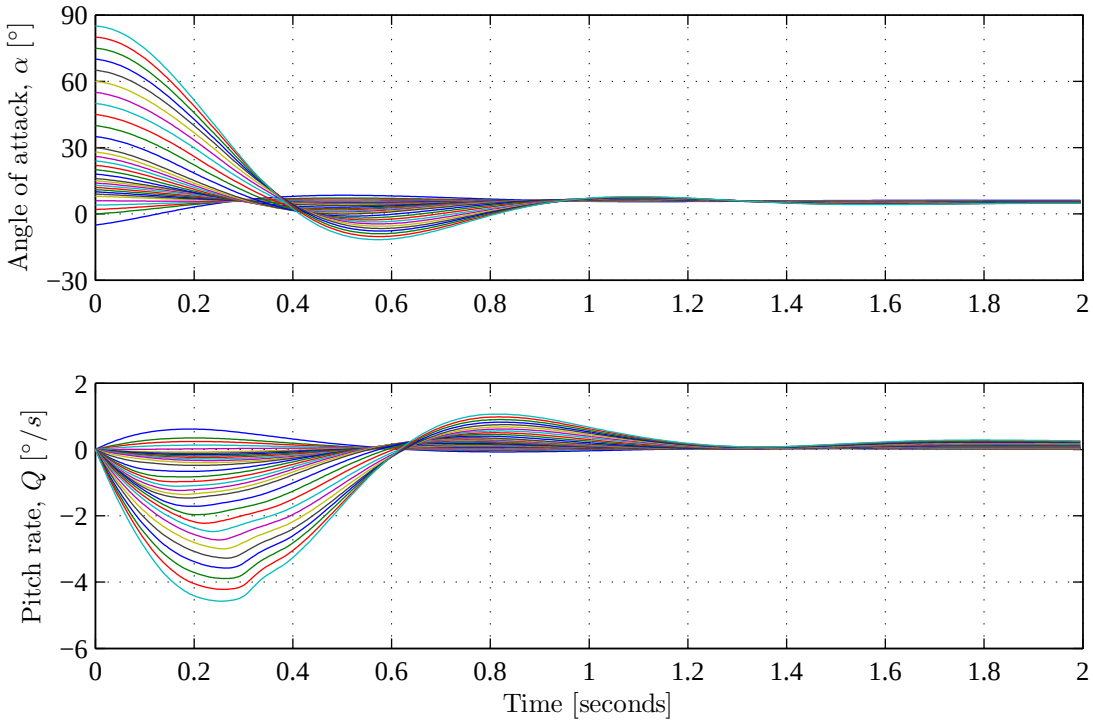


Figure 4.15: Passive angle of attack recovery: time histories of angle of attack and pitch rate

4.5.3 Passive Sideslip Angle Recovery

The region of attraction for passive sideslip angle recovery was explored by simulating the state trajectories from a grid of initial sideslip angles from -45 to +45 degrees, with the initial angle of attack at trim, and with the zero initial angular rates. The elevator deflection was held at a constant zero degrees, and the engine throttle was held fixed at idle. The state trajectories of sideslip angle versus yaw rate are shown in figure 4.16 and the time histories are shown in figure 4.17.

The state trajectories show that all trajectories converge to the stable equilibrium inside the aerodynamic envelope and that no trajectories diverge to infinity. For all initial sideslip angles from -45 to +45 degrees, the sideslip angle returns to the stable equilibrium at $\beta = 0$ degrees. A statistical analysis of the time histories shows that the sideslip angle recovery has an average 5% settling time of 3.5 seconds, with a standard deviation of 1.6 seconds.

The region of attraction for passive sideslip angle recovery therefore spans the range of sideslip angles from -45 to +45 degrees, and the sideslip angle is passively recovered within about 3.5 seconds.

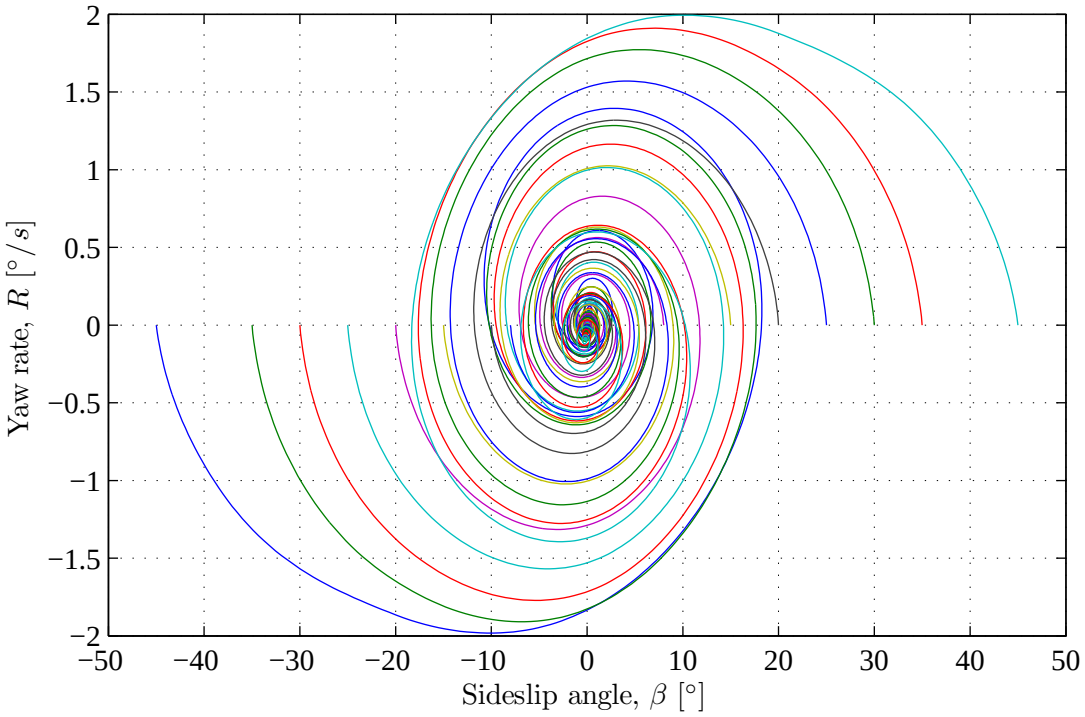


Figure 4.16: Passive sideslip angle recovery: state trajectories of sideslip angle vs. yaw rate

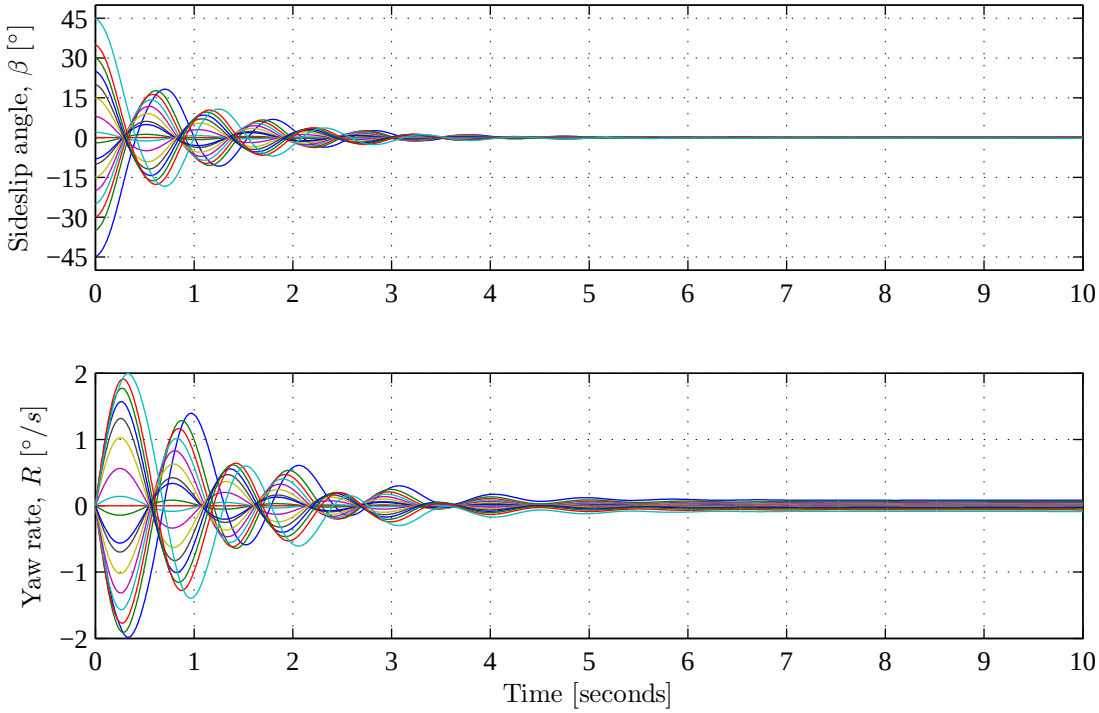


Figure 4.17: Passive sideslip angle recovery: time histories of sideslip angle and yaw rate

4.5.4 Passive Combined Angle of Attack and Sideslip Angle Recovery

The region of attraction for passive combined angle of attack and sideslip angle recovery was explored by simulating the state trajectories from a grid of initial states with angles of attack from -5 to +85 degrees, sideslip angles from -45 to +45 degrees, and with zero initial angular rates. The elevator deflection was held at a constant zero degrees, and the engine throttle was held fixed at idle. The state trajectories of angle of attack versus sideslip angle are shown in figure 4.18 and the time histories are shown in figure 4.19.

The state trajectories show that all trajectories converge to the stable equilibrium inside the aerodynamic envelope and that no trajectories diverge to infinity. For all combinations of initial angles of attack from -5 to +85 degrees and initial sideslip angles from -45 to +45 degrees, the angle of attack and the sideslip angle return to the stable equilibrium at $\alpha = 5.5$ degrees and $\beta = 0$ degrees. A statistical analysis of the time histories shows that the angle of attack has an average 5% settling time of 1.9 seconds, with a standard deviation of 1.8 seconds, while the sideslip angle has an average 5% settling time of 2.9 seconds with a standard deviation of 1.5 seconds.

The region of attraction for passive combined angle of attack and sideslip angle recovery therefore spans the domain of angles of attack from -5 to +85 degrees and sideslip angles from -45 to +45 degrees. The angle of attack is passively recovered within about 1.9 seconds, and the sideslip angle is passively recovered within about 2.9 seconds.

It should be noted that for some trajectories with very high initial angles of attack, the angle of attack of the trajectory goes below -5 degrees, which is outside the lookup table range of the NASA GTM wide-envelope aerodynamic model. For these trajectories the lookup tables are extrapolated calculate the aerodynamic forces and moments.

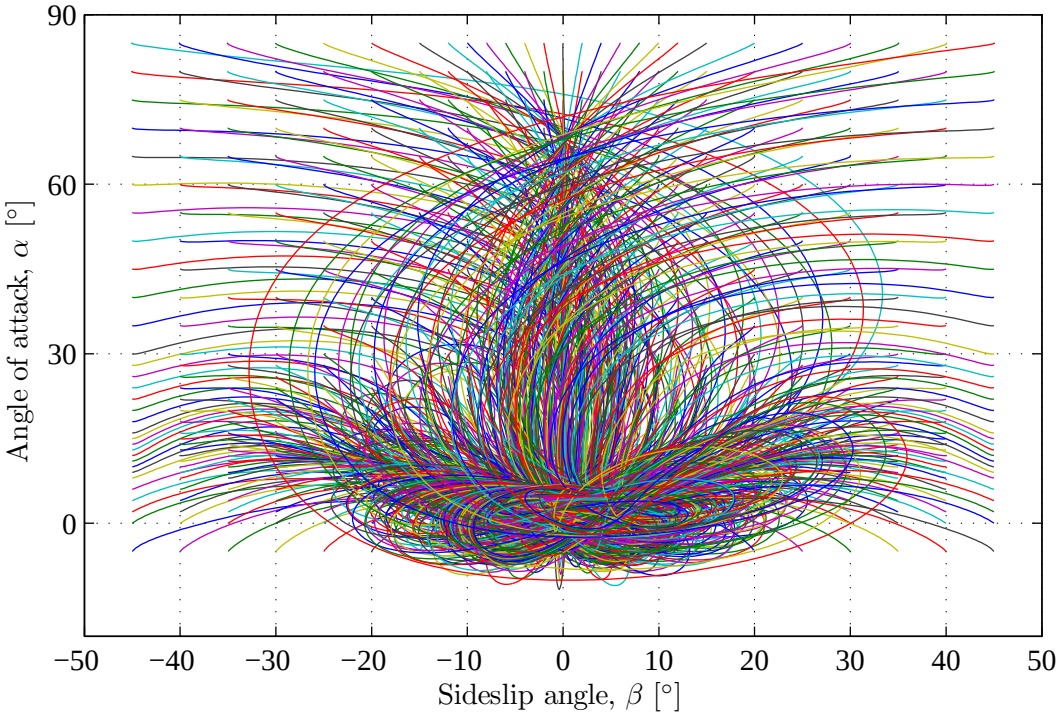


Figure 4.18: Passive angle of attack and sideslip angle recovery: state trajectories of angle of attack vs. sideslip angle

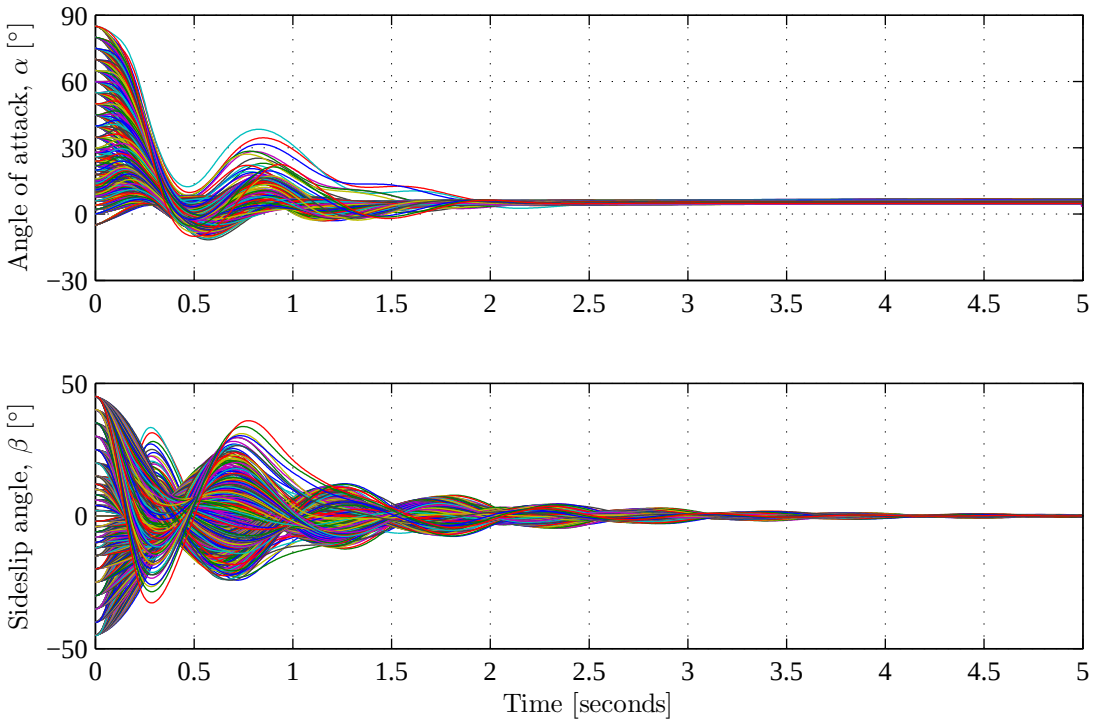


Figure 4.19: Passive angle of attack and sideslip angle recovery: time histories of angle of attack and sideslip angle

4.5.5 Passive Spin Recovery

The region of attraction for passive spin recovery and aerodynamic envelope recovery was determined by performing Monte Carlo simulations where state trajectories were simulated from a comprehensive number of initial states with angles of attack, sideslip angles, and wind-axis roll rates uniformly distributed over the set of valid states for the Generic Transport Model. Monte Carlo simulations were performed with the aircraft state randomly initialised with angles of attack uniformly distributed from -5 to +85 degrees, sideslip angles uniformly distributed from -45 to +45 degrees, and with the aircraft spinning about its velocity vector with wind-axis roll rates uniformly distributed from -180 to +180 degrees per second. All other flight variables were initialised inside the flight envelope (shallow negative flight path angle of -5 degrees, zero bank angle, and a typical airspeed of 75 knots). The control surface deflections and the engine thrust were trimmed for level flight with an equivalent airspeed of 75 knots and were held constant during the simulations. A set of 1000 Monte Carlo simulations were performed.

The time histories of the angle of attack and the sideslip angle are shown in figure 4.20 and the time histories of the body angular rates are shown in figure 4.21. Scatter plots of the initial angles of attack, initial sideslip angles and initial wind-axis roll rates are shown in figures 4.22 and 4.23 to show the coverage of the Monte Carlo simulations.

The time histories show that all trajectories converge to the stable equilibrium inside the aerodynamic envelope and that no trajectories diverge to infinity. The body angular rates are recovered to near zero degrees per second, and the angle of attack and sideslip angle are returned to the stable equilibrium at $\alpha = 5.5$ degrees and $\beta = 0$ degrees for all combinations of initial angles of attack from -5 to +85 degrees, initial sideslip angles from -45 to +45 degrees, and wind-axis roll rates from -180 to +180 degrees per second.

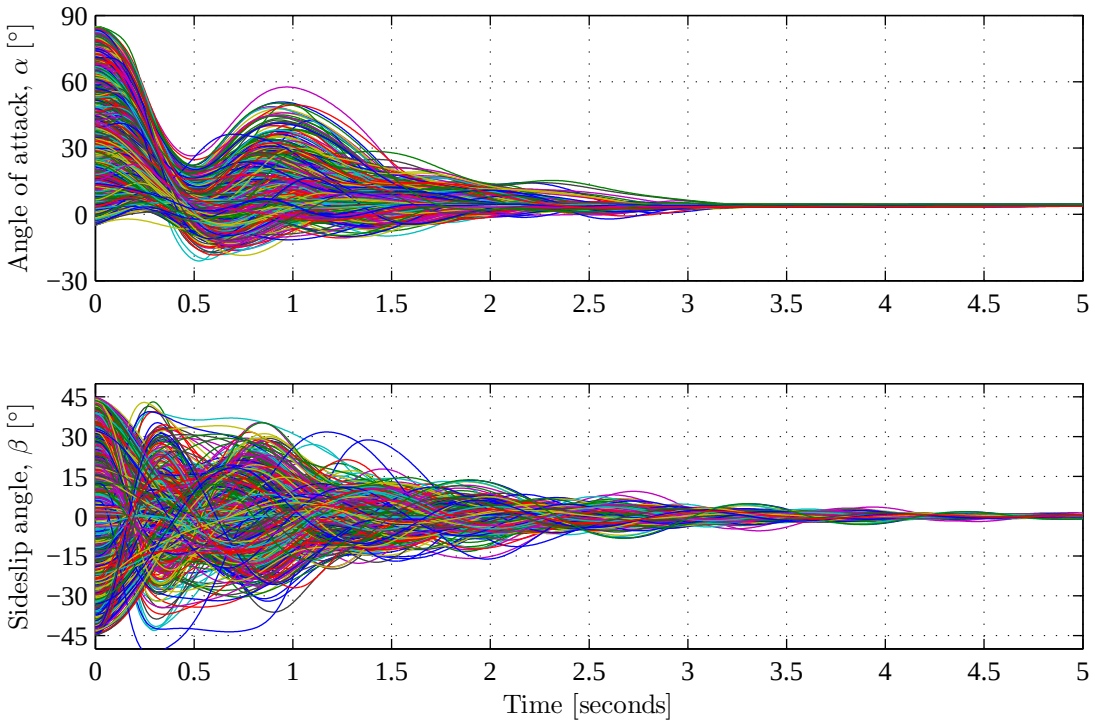


Figure 4.20: Passive spin recovery: time histories of angle of attack and sideslip angle

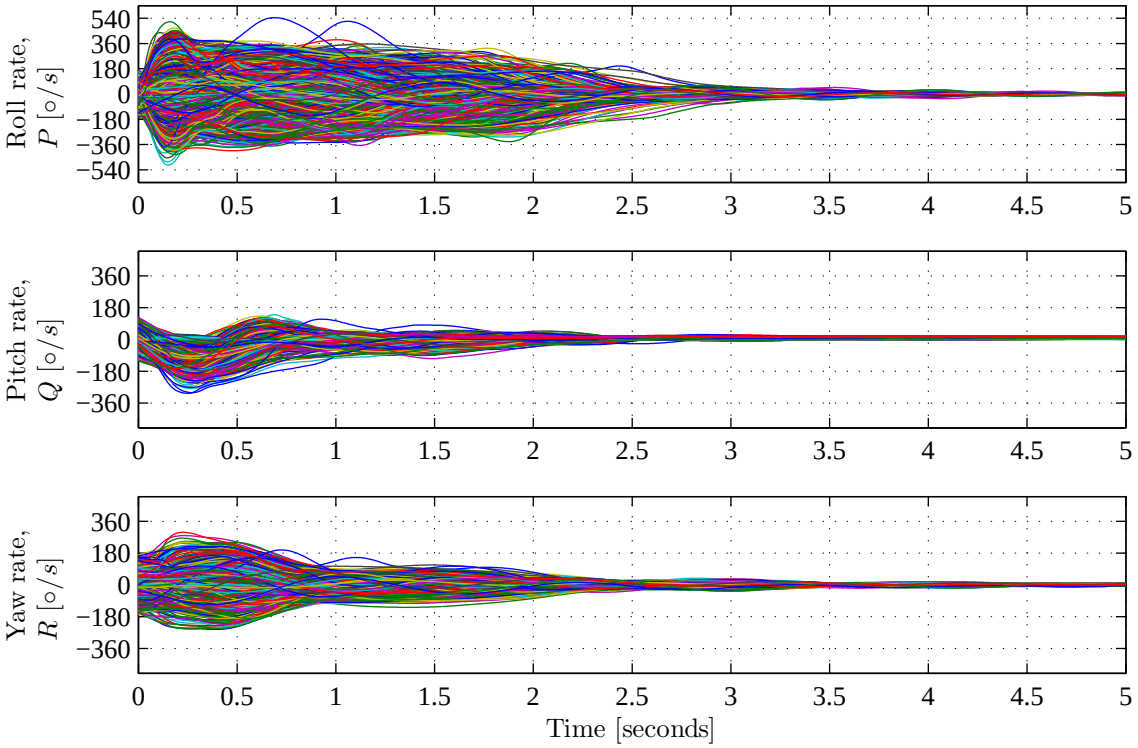


Figure 4.21: Passive spin recovery: time histories of body angular rates

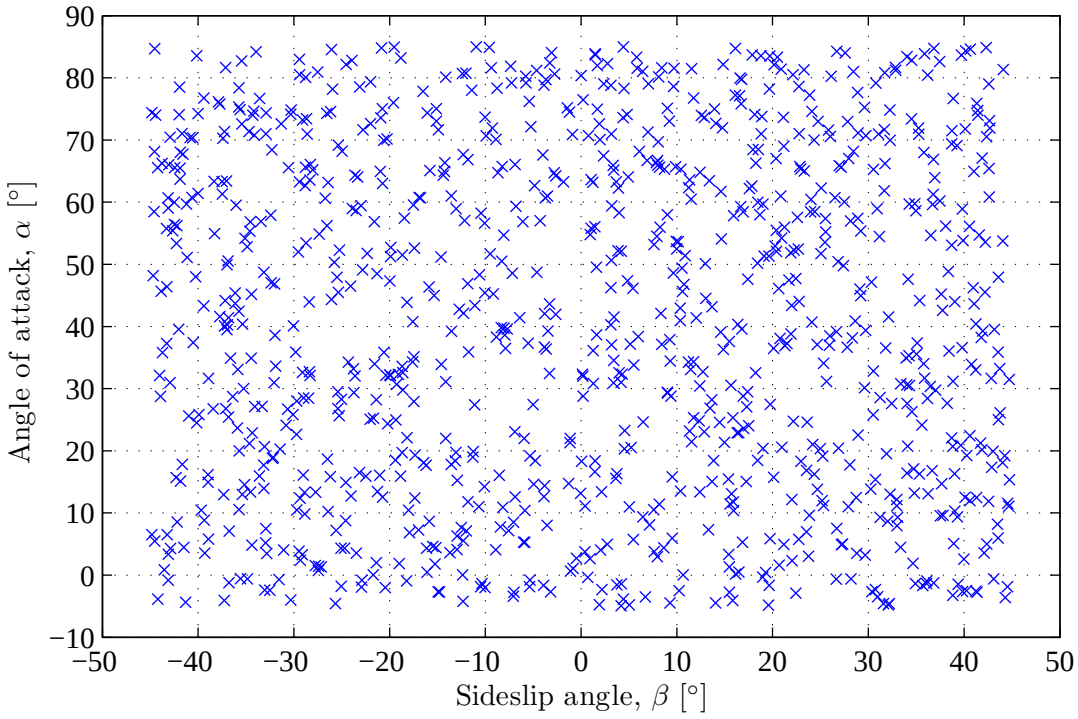


Figure 4.22: Passive spin recovery: scatter plot of initial angle of attack and sideslip angle

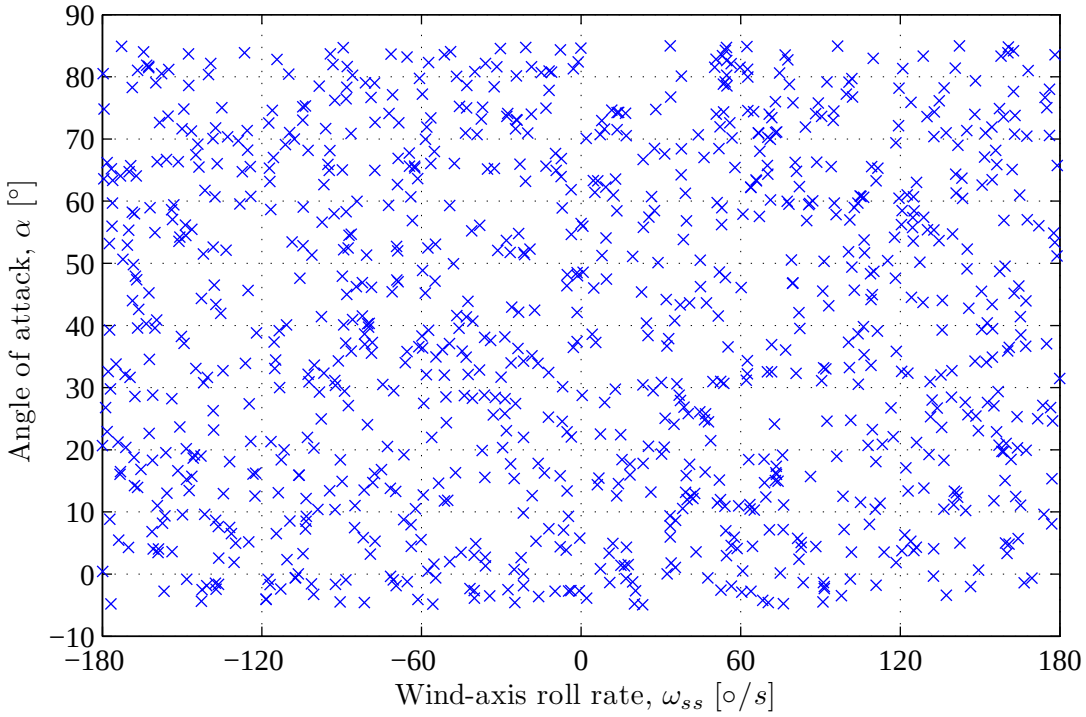


Figure 4.23: Passive spin recovery: scatter plot of initial angle of attack and wind-axis roll rate

4.5.6 Elevator-Induced Spins

For interest sake, the regions of attraction of the undesirable stable spin equilibria outside the aerodynamic envelope were also investigated. The regions of attraction were explored by performing Monte Carlo simulations where state trajectories were simulated from a comprehensive number of initial states, but with the elevator deflection set to -30 degrees. Monte Carlo simulations were performed with the aircraft state randomly initialised with angles of attack uniformly distributed from -5 to +85 degrees, sideslip angles uniformly distributed from -45 to +45 degrees, and with the aircraft spinning about its velocity vector with wind-axis roll rates uniformly distributed from -180 to +180 degrees per second. All other flight variables were initialised inside the flight envelope (shallow negative flight path angle of -5 degrees, zero bank angle, and a typical airspeed of 75 knots). The aileron and rudder deflections and the engine thrust were trimmed for level flight with an equivalent airspeed of 75 knots and were held constant during the simulations. A set of 1000 Monte Carlo simulations were performed.

The time histories of the angle of attack, the sideslip angle and the body angular rates are shown in figures 4.24 and 4.25. The state trajectories of the angle of attack versus the sideslip angle and the wind-axis roll rate versus the sideslip angle are shown in figures 4.26 and 4.27. Scatter plots of the initial angles of attack, initial sideslip angles and initial wind-axis roll rates are shown in figures 4.28, 4.29, 4.30.

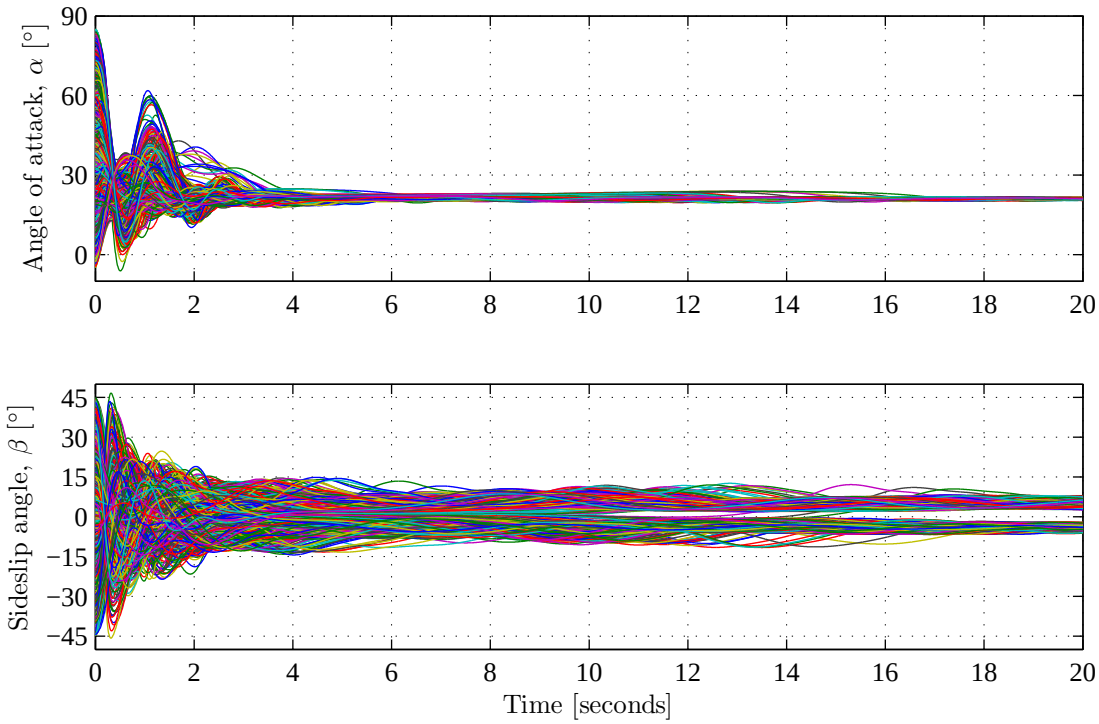


Figure 4.24: Spin entry: time histories of angle of attack and sideslip angle

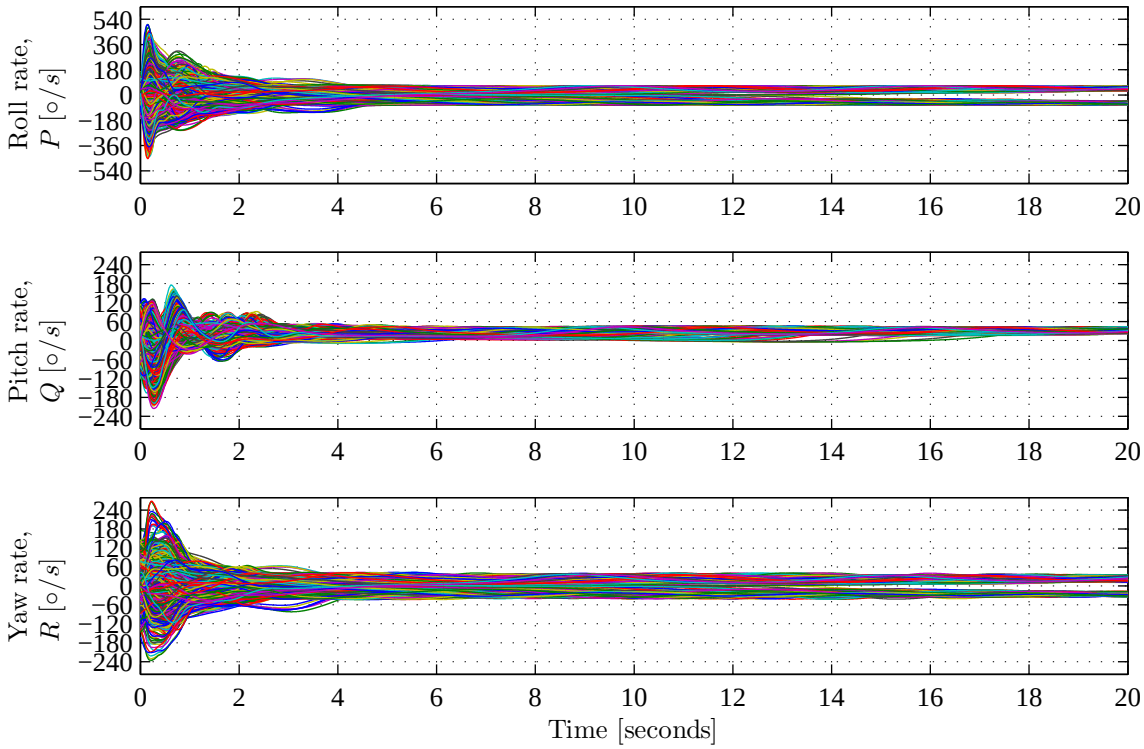


Figure 4.25: Spin entry: time histories of body angular rates

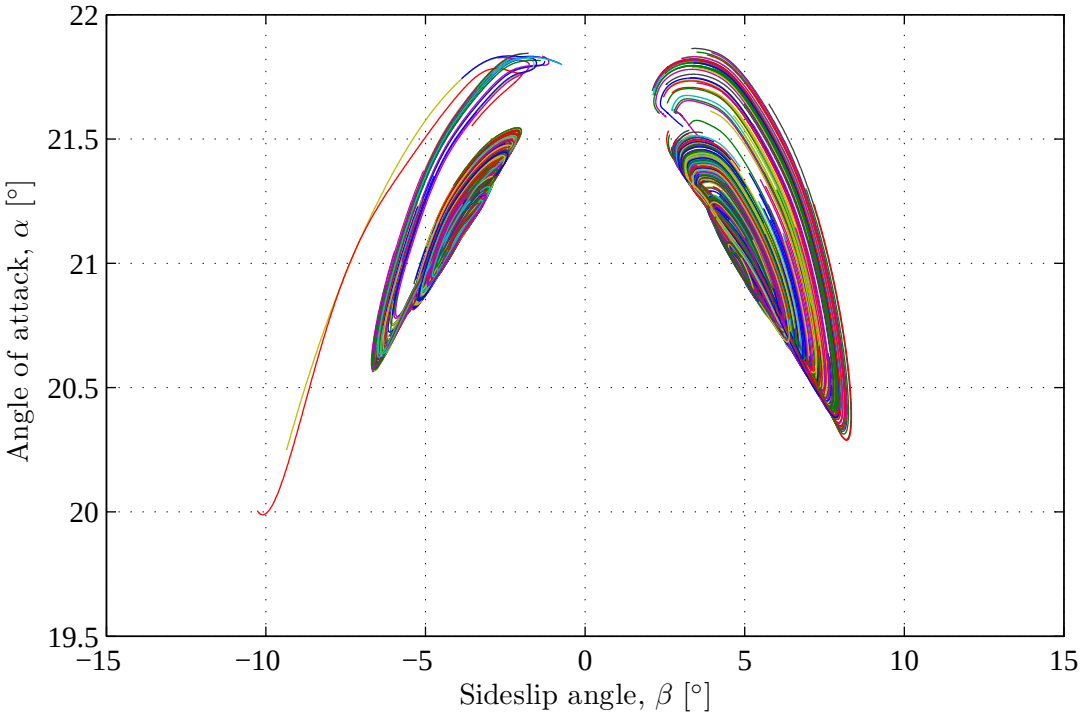


Figure 4.26: Spin entry: state trajectories of angle of attack vs. sideslip angle

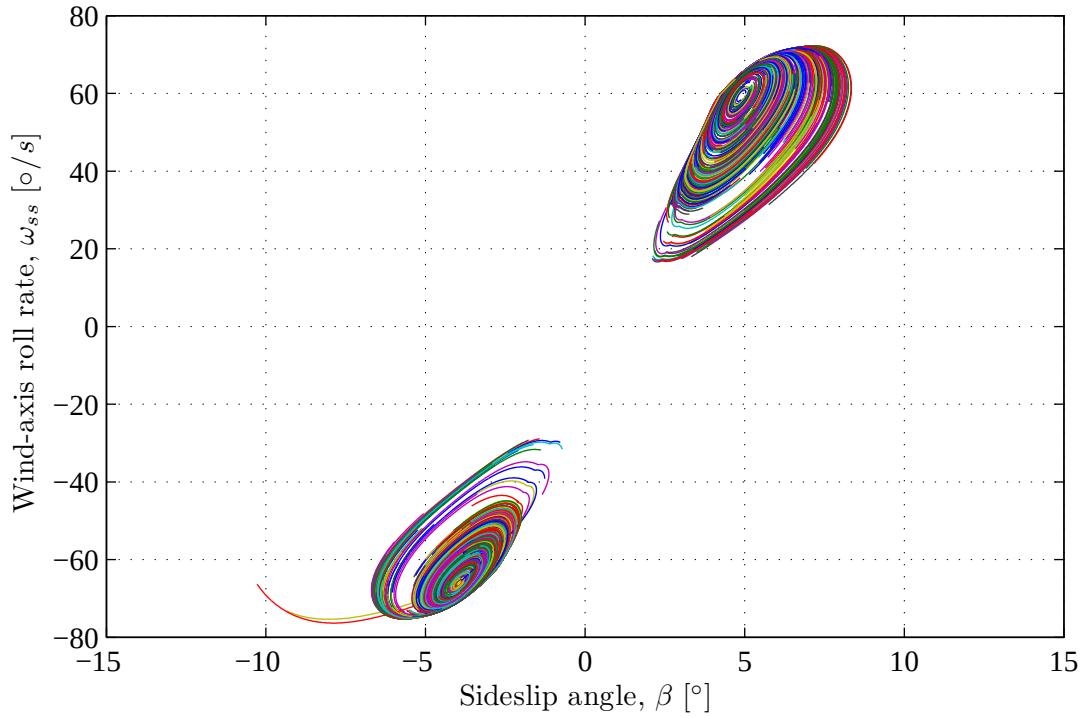


Figure 4.27: Spin entry: state trajectories of wind-axis roll rate vs. sideslip angle.

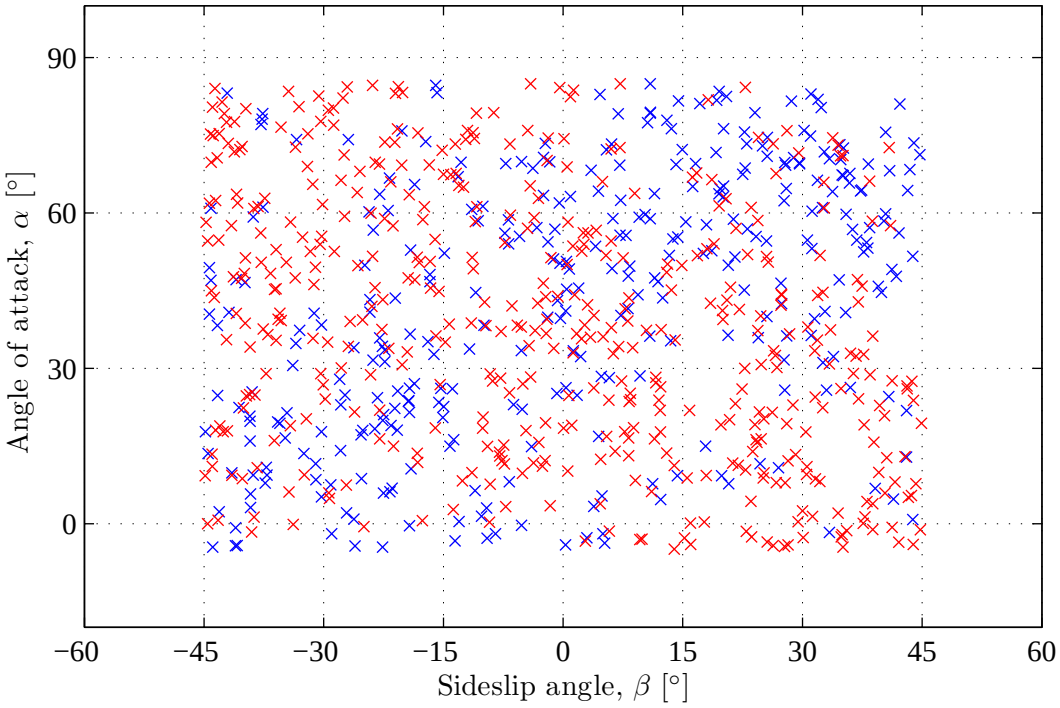


Figure 4.28: Spin entry: scatter plot of initial angle of attack and sideslip angle. (Blue states converge to positive spin. Red states converge to negative spin.)

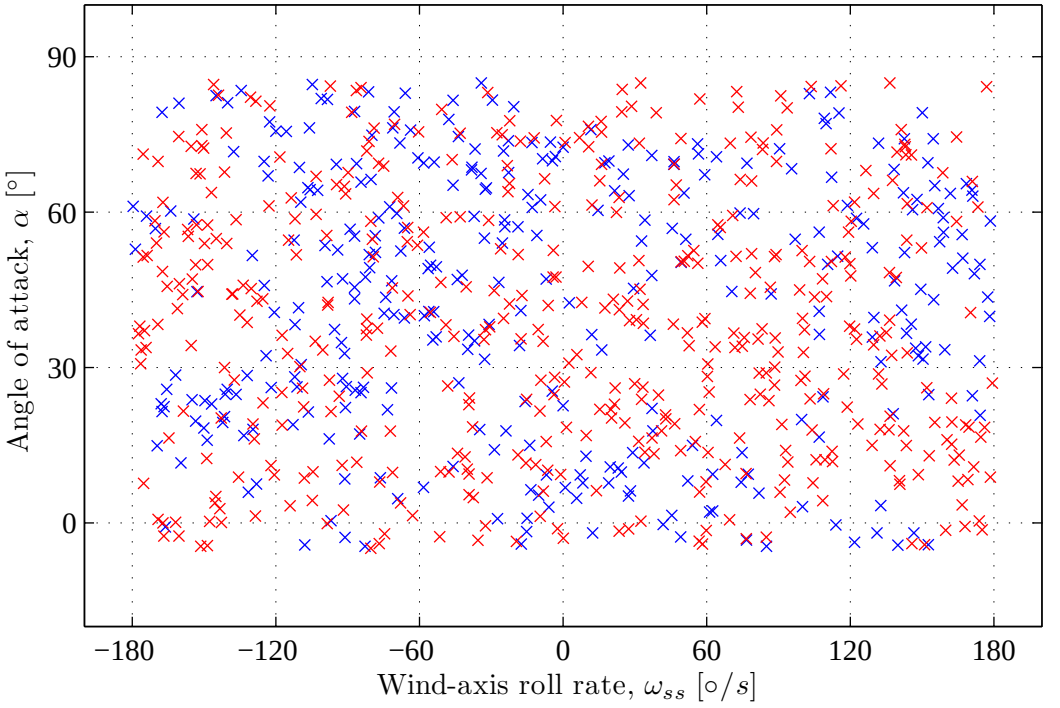


Figure 4.29: Spin entry: scatter plot of initial angle of attack and wind-axis roll rate. (Blue states converge to positive spin. Red states converge to negative spin.)

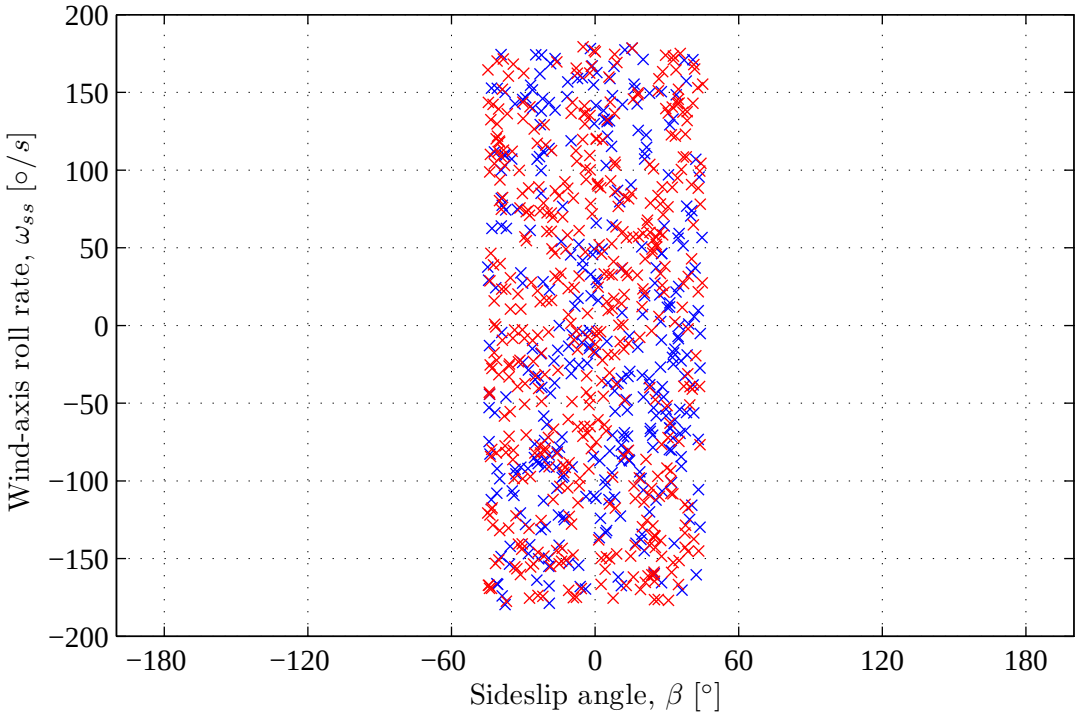


Figure 4.30: Spin entry: scatter plot of initial sideslip angle and wind-axis roll rate. (Blue states converge to positive spin. Red states converge to negative spin.)

The time histories show that all trajectories converge to the two stable spin equilibria outside the aerodynamic envelope and that no trajectories diverge to infinity. By counting the number of trajectories that converge to each of the two equilibria, it was found that 41% of the trajectories converge to the positive spin equilibrium, and 59% of the trajectories converge to the negative spin equilibrium.

The state trajectories visually reveal the basins of attraction for states that start close to each of the two stable spin equilibria. The basin of attaction of the positive spin equilibrium is centred around an angle of attack of 21 degrees, a sideslip angle of 5 degrees, and a wind-axis roll rate of 60 degrees per second. The basin of attaction of the negative spin equilibrium is centred around an angle of attack of 21 degrees, a sideslip angle of -4 degrees, and a wind-axis roll rate of -63 degrees per second.

The scatter plots show which initial states eventually converge to which of the two stable equilibria. The initial states that eventually converge to the positive spin are marked with blue crosses, and the initial states that eventually converge to the negative spin are marked with red crosses. The scatter plots show that for system states that do not start quite close to the equilibrium states, there is no clear correlation between the specific spin equilibrium state that the system trajectory converges to and the initial angle of attack, the initial sideslip angle or the initial spin rate. Both positive and negative initial negative spin states may eventually converge to the positive spin equilibrium, and both positive and negative initial spin states may also eventually converge to the negative spin equilibrium.

4.6 Conclusions

The bifurcation analysis, nonlinear simulations and region of attraction analyses verify that passive angular rate and aerodynamic envelope recovery can be performed by returning the control

surfaces to their neutral deflections and setting the throttle to idle. The steady-state angles of attack, sideslip angles, body angular rates, bank angles and flight path angles observed in the nonlinear simulations correspond well with the equilibrium state predicted by the bifurcation analysis. The region of attraction analyses show that for elevator deflections close to zero all trajectories with initial angles of attack in the range -5 to +85 degrees, sideslip angles in the range -45 to +45 degrees, and wind-axis roll rates in the range -180 to +180 degrees per second converge to the stable equilibrium inside the aerodynamic envelope, and that no trajectories are attracted to other stable equilibria or diverge to infinity. For passive angle of attack recovery, the angle of attack is recovered with an average 5% settling time of 0.8 seconds. For passive combined angle of attack and sideslip angle recovery, the angle of attack is recovered with an average 5% settling time of 1.9 seconds, and the sideslip angle is recovered with an average 5% settling time of 2.9 seconds.

4.7 Summary of Contributions

The following contributions were made in this chapter:

- A passive aerodynamic envelope recovery method was proposed that uses the natural angular rate damping of the aircraft to recover from high angular rates, and uses the natural stability of the aircraft to recover the angle of attack and sideslip angle. The passive approach does not require explicit knowledge of the forces and moments produced by the control surfaces, and does not rely on the availability of wide-envelope anemometric sensor measurements.

- A bifurcation analysis was performed to identify the desirable stable equilibria of the aircraft inside the aerodynamic envelope, and to check for undesirable stable equilibria outside the aerodynamic envelope.

- For control surface deflections near zero, only one stable equilibrium branch was found, corresponding to the aircraft gliding with the steady-state angle of attack and sideslip angle well within the envelope. For large nose-up elevator deflections, two stable branches were found corresponding to stall and spin conditions.

- The regions of attraction of the stable equilibria, both inside and outside the aerodynamic envelope, were explored with nonlinear simulations by generating state trajectories and by performing Monte Carlo simulations.

- The passive aerodynamic envelope approach was verified in simulation on the NASA GTM, and the simulation results show the approach can be used to successfully perform angle of attack recovery, sideslip angle recovery, combined angle of attack and sideslip angle recovery, and stall and spin recovery.

Chapter 5

State Machine Based Attitude and Flight Vector Recovery

This chapter presents a state machine based approach to attitude and flight vector recovery that incorporates the passive aerodynamic envelope recovery presented in chapter 4. The purpose of the state machine based approach is to illustrate that once the aerodynamic envelope has been recovered, the conventional flight control laws and protection functions can be used to recover the attitude, flight path angle, and airspeed. A conventional flight control architecture similar to the architecture of fly-by-wire flight control systems used on commercial passenger aircraft was implemented, and the control laws were designed based on the flight dynamics of the NASA GTM. A state machine was designed to sequentially recover the aerodynamic envelope, the bank angle and flight path angle, and the airspeed. The state machine based attitude and flight vector recovery was then demonstrated in simulation on the full NASA GTM. The research presented in this chapter has been published by the author in Engelbrecht et al [60].

5.1 Approach

The state machine based flight envelope recovery uses the following approach. The first step is to recover the angular rates and aerodynamic envelope using the passive aerodynamic envelope recovery described in chapter 4. Once the aerodynamic envelope has been recovered, the conventional flight control laws and protection functions are available to be used for the next stages of bank angle recovery, flight path angle recovery, and airspeed recovery. A bank angle controller with an inner-loop roll rate controller is used to return the bank angle to wings level, and a flight path angle controller with an inner-loop normal load factor controller is used to recover the flight path angle to level flight. A positive flight path angle is used to recover from overspeed using gravity assistance. During the recovery, the angle of attack protection function prevents the aircraft from exiting the aerodynamic envelope again, and the load factor protection function keeps the normal load factor within safe limits.

5.2 Conventional Flight Control Architecture

The state machine and conventional flight control architecture shown in figure 5.1 was implemented for the NASA GTM. The flight control architecture was designed to be similar to the architecture of fly-by-wire flight control systems used on current large commercial passenger aircraft.

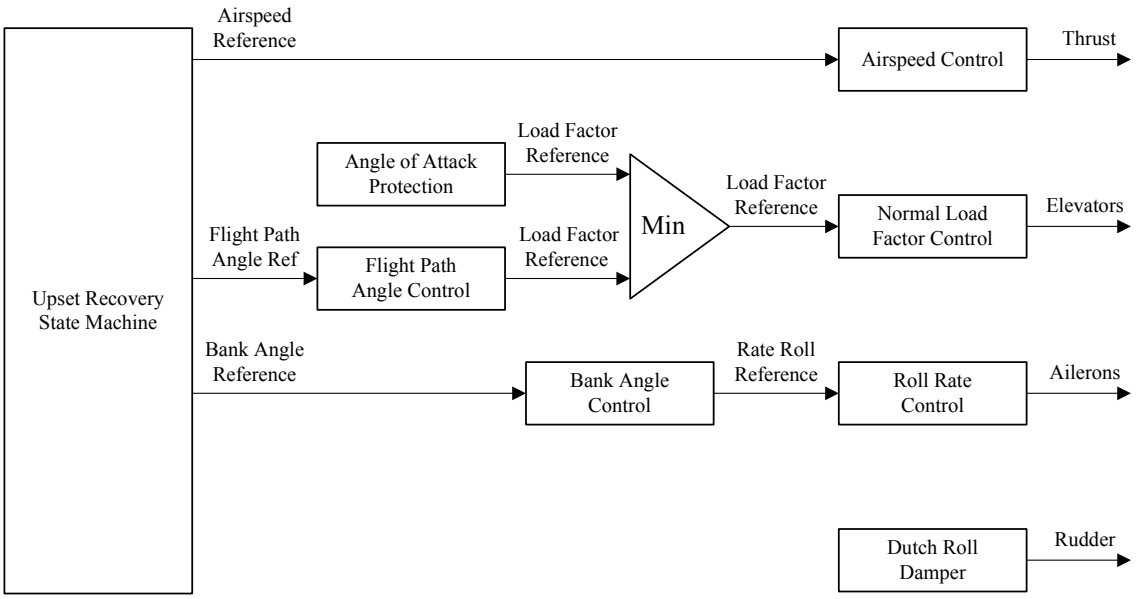


Figure 5.1: State machine and conventional flight control laws used for flight envelope recovery

The architecture and design of the fly-by-wire flight control system and the flight envelope protection functions are proprietary to the manufacturers of large commercial airliners, and not much information is available in the open literature. However, some high-level information on the general flight control architecture, the types of flight control laws used, and the types of flight envelope protections implemented, was obtained from an article by Favre [10] that presented an overview of the Airbus fly-by-wire control laws used on the Airbus A320 and the Airbus A340. The conventional flight control system used in this chapter was not designed to accurately duplicate a specific fly-by-wire control system, but rather to use the same general architecture, and to implement the same types of control laws and protection functions.

The conventional flight control laws use feedback from onboard sensors that are typically available on a large transport aircraft. It is assumed that an inertial measurement system provides three-axis linear acceleration measurements, three-axis angular rate measurements, estimated attitude angles, and GPS velocity and position, and that anemometric sensors provide measurements of the airspeed, angle of attack, sideslip angle, and dynamic pressure.

The airspeed is controlled by an airspeed controller that uses feedback from the airspeed measured by the anemometric sensors to actuate the engine thrust. The flight path angle is controlled by an outer-loop flight path angle controller with an inner-loop normal load factor controller. The inner-loop normal load factor controller uses feedback from the normal load factor measured by the inertial sensors to actuate the elevator deflection, and the outer-loop flight path angle controller uses feedback from the estimated flight path angle obtained from the GPS velocity to command the normal load factor reference for the inner-loop normal load factor controller. The bank angle is controlled by an outer-loop bank angle controller with an inner-loop roll rate controller. The roll rate controller uses feedback from the roll rate measured by the inertial sensors to actuate the aileron deflection, and the outer-loop bank angle controller

uses feedback from the estimated bank angle provided by the inertial measurement system to command the roll rate reference for the inner-loop bank angle controller. A Dutch roll damper is used to improve the damping of the relatively underdamped natural Dutch roll mode of the aircraft by commanding rudder deflections proportional to the yaw rate measured by the inertial sensors.

Only the flight envelope protection functions that would be used for the flight envelope recovery, namely load factor limiting and angle of attack protection, were implemented. Airspeed protection, flight path angle protection, and bank angle protection functions were not implemented, since these were the flight variables that would be recovered by the attitude and flight vector recovery state machine.

The state machine controls which of the control laws are active and provides the references for the airspeed controller, the flight path angle controller, and the bank angle controller based on the current recovery state. A brief overview of the architecture and design methodology of each control law is provided in the sections below.

5.2.1 Airspeed Controller

The block diagram of the airspeed controller is shown in figure 5.2. The airspeed controller is a simple proportional controller that uses feedback from the airspeed sensors and actuates the engine throttle. A saturation block was used to limit the throttle command so that it does not exceed the throttle range of the aircraft.

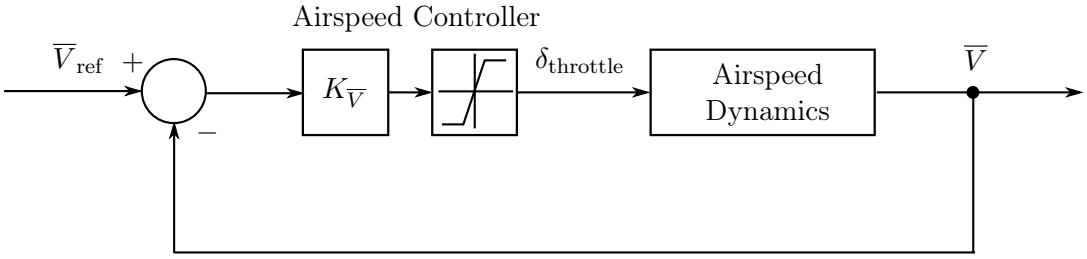


Figure 5.2: Airspeed controller architecture

The proportional gain for the airspeed controller was designed based on a second-order reduced model of the airspeed dynamics. The longitudinal dynamics was linearised about the nominal trim condition with an equivalent airspeed of 75 knots and an altitude of 800 feet. The fourth-order longitudinal state space model was then reduced to a first-order state space model with airspeed as the state and throttle as the input. The natural airspeed dynamics was found to be quite slow with a time constant of 20 seconds. The first-order state space model was converted to a transfer function and augmented with a first order pole with a time constant of 2.36 seconds, representing the throttle lag dynamics. (The time constant of the throttle lag dynamics was determined experimentally from the simulation model.) A root locus design was then performed to calculate the required proportional gain to provide a closed-loop response that is dominantly first-order and has a time constant of 5 seconds.

5.2.2 Normal Load Factor Controller

The block diagram of the inner-loop normal load factor controller is shown in figure 5.3. The normal load factor controller is a full state feedback controller with integral control that uses feedback from the normal load factor sensor and the pitch rate sensor and actuates the elevators.

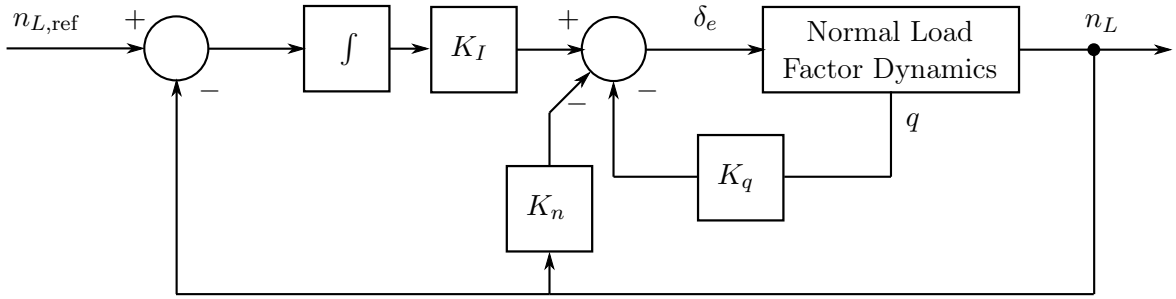


Figure 5.3: Normal load factor controller architecture

The integral term ensures that the normal load factor follows the command with zero steady-state error, while the pitch rate feedback provides artificial damping of the short period mode dynamics. The normal load factor controller inherently provides load factor protection, since it regulates the normal load factor to follow the reference. The normal load factor reference is limited to remain within safe limits.

The normal load factor controller gains were designed using a reduced-order model of the normal load factor dynamics. The longitudinal dynamics were linearised about the nominal trim condition with an equivalent airspeed of 75 knots and an altitude of 800 feet. The fourth-order longitudinal state space model was then reduced to a second-order state space model with normal load factor and pitch rate as the two states and the elevator deflection as the input. The natural normal load factor dynamics were found to be relatively underdamped with the open-loop poles having a damping ratio of $\zeta = 0.44$ and a natural frequency of $w_n = 6.6$ rad/s.

The second-order state space model was then augmented with the integral of the normal load factor error, and pole placement was used to place the closed-loop poles. The closed-loop poles corresponding to the short period mode were placed at the same natural frequency but with the damping improved to $\zeta = 0.9$, and the closed-loop pole corresponding to the integrator was placed to have a time constant of 0.5 seconds.

5.2.3 Flight Path Angle Controller

The block diagram of the outer-loop flight path angle controller is shown in figure 5.4. The flight path angle controller is a proportional controller that uses feedback from the estimated flight path angle and commands the normal load factor reference of the inner-loop normal load factor controller. A saturation block is used to limit the normal load factor reference to remain within safe limits.

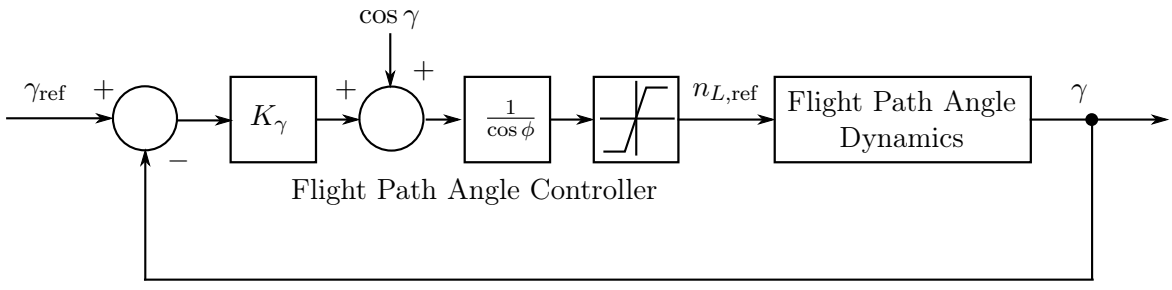


Figure 5.4: Flight path angle controller architecture

The premise of the flight path angle controller is that the flight path angle rate equals the total vertical acceleration perpendicular to the velocity vector, divided by the velocity. The total vertical acceleration perpendicular to the velocity vector is the sum of the normal load factor projected into the vertical plane and the component of gravity perpendicular to the velocity vector. This is expressed mathematically by the following equation

$$\dot{\gamma} = \frac{g}{\overline{V}}(n_L \cos\Phi - \cos\gamma) \tag{5.1}$$

where $\dot{\gamma}$ is the flight path angle rate, g is gravitational acceleration, $\overline{V}$ is the velocity magnitude, n_L is the normal load factor, Φ is the bank angle, and γ is the flight path angle.

The flight path angle controller commands the normal load factor reference to make the flight path angle rate proportional to the flight path angle error, resulting in the following control law

$$n_{L\text{ref}} = \frac{1}{\cos\Phi}[K_\gamma(\gamma_{\text{ref}} - \gamma) + \cos\gamma] \tag{5.2}$$

where n_L is the normal load factor reference, K_γ is the proportional controller gain, and γ_{ref} is the flight path angle reference.

The normal load factor reference is therefore increased as the bank angle is increased, to compensate for the fact that the lift vector is directed away from the vertical. For bank angles close to $\pm$ 90 degrees, the lift vector is in the horizontal plane, and cannot be used to affect the flight path angle rate. For bank angles greater than $\pm$ 90 degrees, the lift vector starts pointing downwards and a positive normal load factor produces a negative flight path angle rate. Control logic was therefore implemented so that the calculated normal load factor reference is only used for bank angles up to $\pm$ 80 degrees. For bank angles greater than $\pm$ 80 degrees, the normal load factor reference is made zero.

The proportional gain for the flight path angle controller was designed based on a reduced-order model of the flight path angle dynamics. The transfer function from the total vertical acceleration to the flight path angle was modelled as an integrator with a gain of $g/\overline{V}$. The transfer function was augmented with a first order pole with a time constant of 0.5 seconds representing the dynamics of the normal load factor controller. A root locus design was then performed to calculate the required proportional gain that would provide an optimally damped closed-loop response with a settling time of 4 seconds.

5.2.4 Angle of Attack Protection

An angle of attack protection function is implemented in parallel with the flight path angle controller. The block diagram of the angle of attack protection controller is shown in figure 5.5. It is a simple proportional controller that uses feedback from the angle of attack sensor and actuates the normal load factor command. The angle of attack reference is always set to the maximum allowable angle of attack. A saturation block is used to limit the normal load factor reference to remain within safe limits.

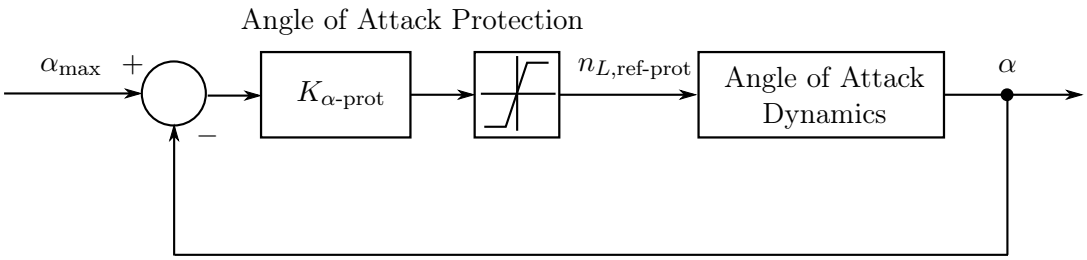


Figure 5.5: Angle of attack protection controller architecture

The flight path angle controller and the angle of attack protection function both output normal load factor commands, but only the most nose-down of the two commands is actually passed to the normal load factor controller, as shown in figure 5.6. As long as the flight path angle controller commands more nose-down normal load factor than the angle of attack protection function, it will be the active controller. However, as soon as the angle of attack controller commands more nose-down normal load factor than flight path angle controller, the protection function will be the active controller. The minimum function provides a simple, passive selection mechanism, with smooth transitions between the control law and the protection function, and without complex decision logical.

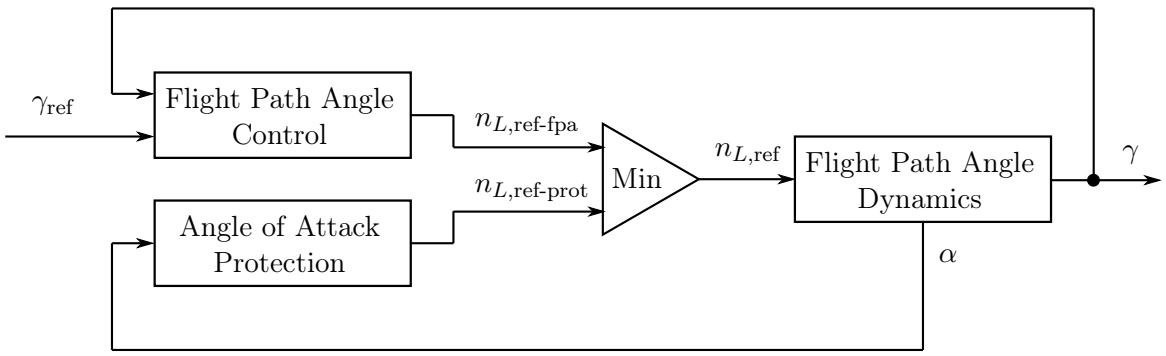


Figure 5.6: Activation of angle of attack protection

5.2.5 Roll Rate Controller

The block diagram of the inner-loop roll rate controller is shown in figure 5.7. The roll rate controller is an integral controller that uses feedback from the roll rate sensor and actuates the ailerons. The controller also has a direct feedforward gain from the roll rate reference to the aileron command. A saturation block is used to limit the aileron command so that it does not exceed the aileron deflection range of the aircraft. A limited integrator is used to prevent integrator wind-up.

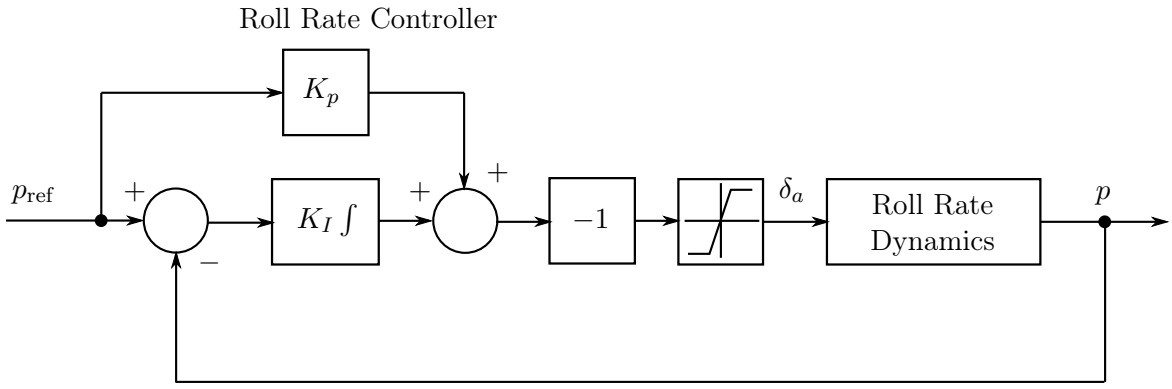


Figure 5.7: Roll rate controller architecture

The integral term ensures that the measured roll rate follows the reference roll rate with zero steady-state error. The direct feedforward term is used to speed up the transient response by compensating for the lag dynamics introduced by the integrator.

The gains for the integral controller with direct feedforward gain were designed based on a reduced-order model of the roll mode dynamics. The roll mode dynamics are well approximated by a first-order model with roll rate as output and aileron deflection as input. A root locus design was performed to calculate the required integral gain to place the closed-loop pole corresponding to the roll mode dynamics at a frequency of 5 rad/s. The feedforward gain was then calculated to place a zero on top of the slower closed-loop pole corresponding to the integrator, making the faster closed-loop pole dominant. The resulting closed-loop response is dominantly first-order and has a time constant of 0.2 seconds.

5.2.6 Bank Angle Controller

The block diagram of the outer-loop bank angle controller is shown in figure 5.8. The bank angle controller is a simple proportional controller that uses feedback from the estimated bank angle to actuate the roll rate command. A saturation block is used to limit the roll rate command given to the inner-loop roll rate controller.

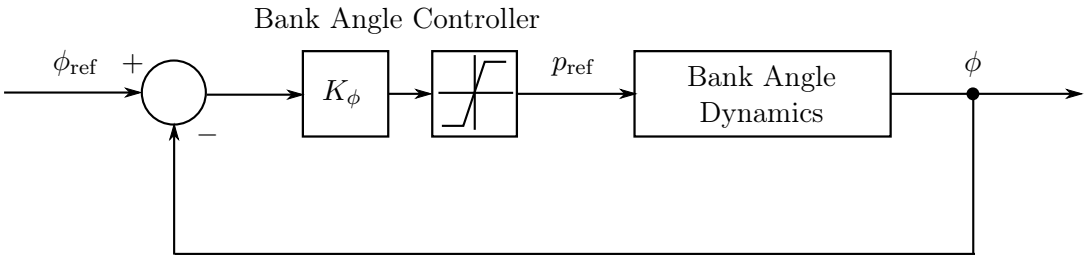


Figure 5.8: Bank angle controller architecture

The proportional gain for the bank angle controller was designed based on a reduced-order model of the roll angle dynamics. The reduced order model starts with the approximation that the bank angle is simply the integral of the roll rate. The transfer function is then augmented with a first order pole with a time constant of 0.2 seconds representing the dynamics of the

inner-loop roll rate controller. A root locus design was then performed to calculate the required proportional gain that would provide a dominantly first-order closed-loop response with a time constant of 1 second.

5.2.7 Dutch Roll Damper

The block diagram of the Dutch roll damper is shown in figure 5.9. The Dutch roll damper uses feedback from the yaw rate sensor to actuate the rudder deflection.

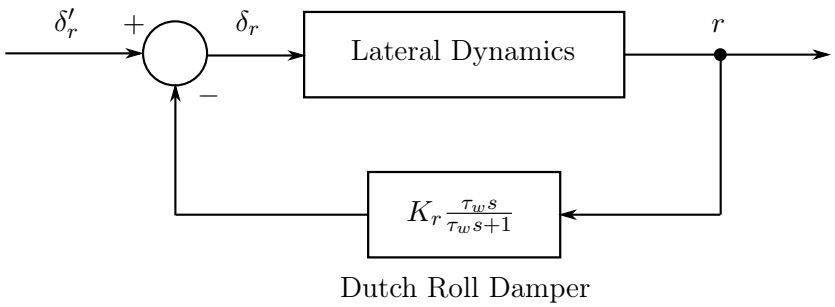


Figure 5.9: Dutch roll damper architecture

The Dutch roll mode of the GTM is found to be quite lightly damped, and an inner-loop Dutch roll damper was implemented to add artificial damping to the Dutch roll mode in order to suppress the natural elliptical motion of the aircraft tail when it is disturbed. To add artifial yaw damping, the rudder deflection is actuated proportionally to the yaw rate measured by the inertial sensors. The yaw rate signal is fed through a high pass filter, or washout filter, so that damping is only provided for higher frequency yaw oscillations, and so that the rudder is not deflected in response to constant yaw rates, such as those experienced during normal turn maneuvers. The filter cut-off frequency is chosen to allow the higher frequency Dutch roll mode yawing motions to pass through and be fed back to the rudder, and low enough to sufficiently block the lower frequency yaw rates associated with normal turn maneuvers.

The design of the Dutch roll damper consists of choosing values for the washout filter cutoff frequency $1/\tau_w$ and the proportional feedback gain K_r. Choosing the washout filter cutoff frequency is typically an iterative process. The cutoff frequency must be low enough so that the frequency of the Dutch roll mode lies in the passband without too much phase error, and high enough so as not to interfere with constant turn rate motions. A common approach is to choose the washout filter cutoff frequency equal to a quarter of the natural frequency of the Dutch roll mode as a starting point. Once the washout filter cutoff frequency has been chosen, the proportional gain of the Dutch roll damper is calculated by performing a root locus design and choosing the filter gain that provides sufficient damping for the Dutch roll mode.

The natural Dutch roll mode dynamics of the NASA GTM was found to be quite lightly damped with a damping ratio of $\zeta = 0.15$ and a natural frequency of $w_n = 6.05$ rad/s. The washout filter cutoff frequency was chosen as 1.5 rad/s and the proportional gain was designed to improve the damping ratio to $\zeta = 0.7$.

5.3 Controller Activation per Recovery State

The flight envelope recovery state machine activates the appropriate controllers and assigns appropriate control references as a function of the current flight envelope recovery mode, as summarized in table 5.1.

Table 5.1: Controllers activated and controller references per recovery mode

Control / State	Auto Thrust	NSA Control	Flight Path Angle Control	Roll Rate Control	Bank Angle Control
Recover Aerodynamic Envelope	No $\Delta T = 0$	No $\delta_E = 0$	No	No $\delta_A = 0$	No
Recover Attitude Envelope	$V_{ref} = 75$ kts	Yes	$\gamma_{ref} = 0$ deg	Yes	$\phi_{ref} = 0$ deg
Recover Overspeed			$\gamma_{ref} = 15$ deg		
Recovery Done			$\gamma_{ref} = 0$ deg		

In Recover Aerodynamic Envelope mode, no feedback controllers are active, and instead the actuators are set to fixed values that will allow the natural stability of the aircraft to damp the angular rates and recover the aerodynamic envelope. The throttle is set to idle, and the elevator, ailerons and rudder are set to their neutral deflections. Once the sensors indicate that the angle of attack and sideslip angle have returned to the normal aerodynamic envelope, the state machine transitions to Recover Attitude Envelope mode.

In Recover Attitude Envelope mode, all conventional control laws are activated. The airspeed controller is activated and given a reference of 75 knots. The roll rate and bank angle controllers are activated, and the bank angle reference is set to zero, to recover the bank angle of the aircraft to wings level. The normal load factor controller and flight path angle controllers are activated, and the flight path angle reference is set to zero, to recover the flight path angle of the aircraft to level flight. The normal load factor controller provides load factor protection to keep the normal load factor within safe limits. The angle of attack protection is also activated to prevent the aircraft from exiting the aerodynamic envelope again during the attitude recovery. Once the attitude estimator indicates that the bank angle and flight path angle have returned to the normal attitude envelope, the state machine transitions to Recover Overspeed mode.

In Recover Overspeed mode, the flight path angle reference is set to a positive value, e.g. 15 degrees, to use gravity to rapidly reduce the airspeed. Once the airspeed sensors indicate that the aircraft speed has returned to the normal flight envelope, the state machine transitions to Recovery Done mode. In Recovery Done mode, the flight path angle reference is set to zero again, so that the aircraft returns to straight and level flight. The airspeed controller, which is still active, will automatically increase the engine throttle to maintain the airspeed. The flight path angle controller will maintain level flight, and the bank angle controller will maintain wings level. The automatic recovery procedure is now complete, and the pilot may decide to regain altitude or change heading.

5.4 Simulation Results

A simulation was performed to demonstrate the ability of the state machine based flight envelope recovery system to recover the aircraft from a severe upset condition. The simulation was performed with the full NASA GTM simulation model. The aircraft was trimmed for straight and level flight at an airspeed of 75 knots and an altitude of 8000 feet. The flight envelope recovery system was initially disarmed. At 5 seconds into the simulation, the elevator was deflected to -30 degrees to induce stall and post-stall spin. At 25 seconds into the simulation, the flight envelope recovery system was activated.

The simulation time histories are shown in figures 5.10 to 5.14. During the first 5 seconds, the aircraft maintains normal flight with an angle of attack of about 7 degrees and a sideslip angle near zero. When the elevator is set to -30 degrees deflection at t = 5 seconds, the aircraft stalls to an angle of attack of 21 degrees and a sideslip angle of 4 degrees. The aircraft rolls to an average bank angle of 50 degrees and enters a stable spin in a steep downward spiral with a flight path angle of about -68 degrees. The spin is observed in the body angular rates. The equilibrium values of the angular rates in the simulation correspond well with the angular rates predicted by the bifurcation analysis performed in Chapter 4.

When the flight envelope recovery flight control system is activated at t = 25 seconds, it detects that the angular rates, angle of attack and sideslip angle are all out-of-envelope, and immediately switches to Recover Aerodynamic Envelope mode. The time histories in figures 5.10 and 5.11 show that the angular rates are quickly reduced to zero, and that the angle of attack and sideslip angle are both returned to the aerodynamic envelope within about 1 second.

At t = 26 seconds, the flight envelope recovery system detects that the angular rates, angle of attack and sideslip angle have all returned to envelope, and it switches to Recover Attitude Envelope mode. The time histories in figure 5.12 show that the bank angle controller recovers the aircraft to wings level, and that the flight path angle controller pulls up the aircraft as hard as the angle of attack protection and load factor protection allows.

At t = 30 seconds, the flight envelope recovery flight control system detects that the bank angle and the flight path angle have both returned to the attitude envelope, and switches to Recover Overspeed mode. At this stage the airspeed has increased to 120 knots, and needs to be reduced. A positive flight path angle is therefore commanded to allow gravity to decelerate the aircraft. The time history shows that the aircraft continues to pull up until it reaches and then maintains an ascending flight path angle of 15 degrees. The time history in figure 5.13 shows that the airspeed decreases and returns to the nominal 75 knots within about 10 seconds. The angle of attack steadily increases to compensate for the lift lost due to the decrease in airspeed, so that the flight path angle can be maintained.

At t = 40 seconds, the flight envelope recovery system detects that the airspeed has returned to the normal flight envelope, and switches to Recovery Done mode. The flight path angle is therefore set to zero and the aircraft returns to level flight. A transient is observed in the angle of attack, and it returns to the trim value of 7 degrees expected for straight and level flight. The time history in figure 5.14 shows that the normal load factor remains between 0.5 and 2.5 g during the entire flight envelope recovery procedure, which is within the safe limits for structural integrity and passenger safety. Looking at the time histories for altitude and airspeed in figure 5.14, it is observed that after the automatic flight envelope recovery is activated, the total altitude loss is only 700 feet, and a maximum overspeed of 120 knots was observed during the recovery.

The flight trajectory of the aircraft is shown in figure 5.15 and clearly shows the initial straight and level flight, followed by the steep downward spiral of the post-stall spin, and finally the complete flight envelope recovery, returning the aircraft to straight and level flight.

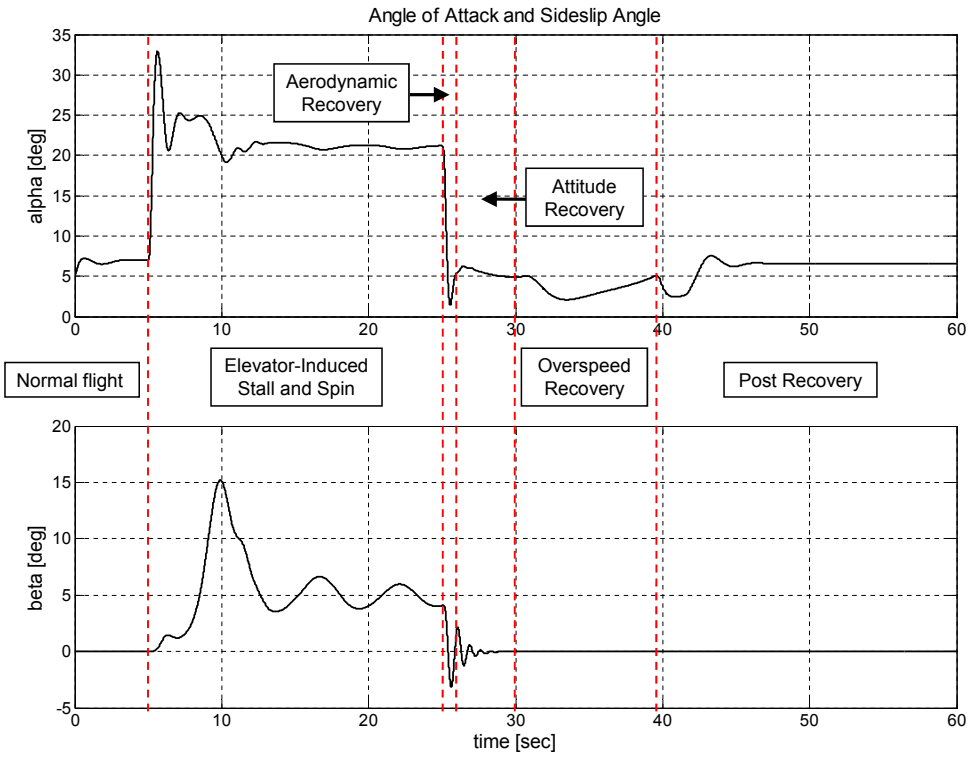


Figure 5.10: State machine based stall and spin recovery: angle of attack and sideslip angle vs. time

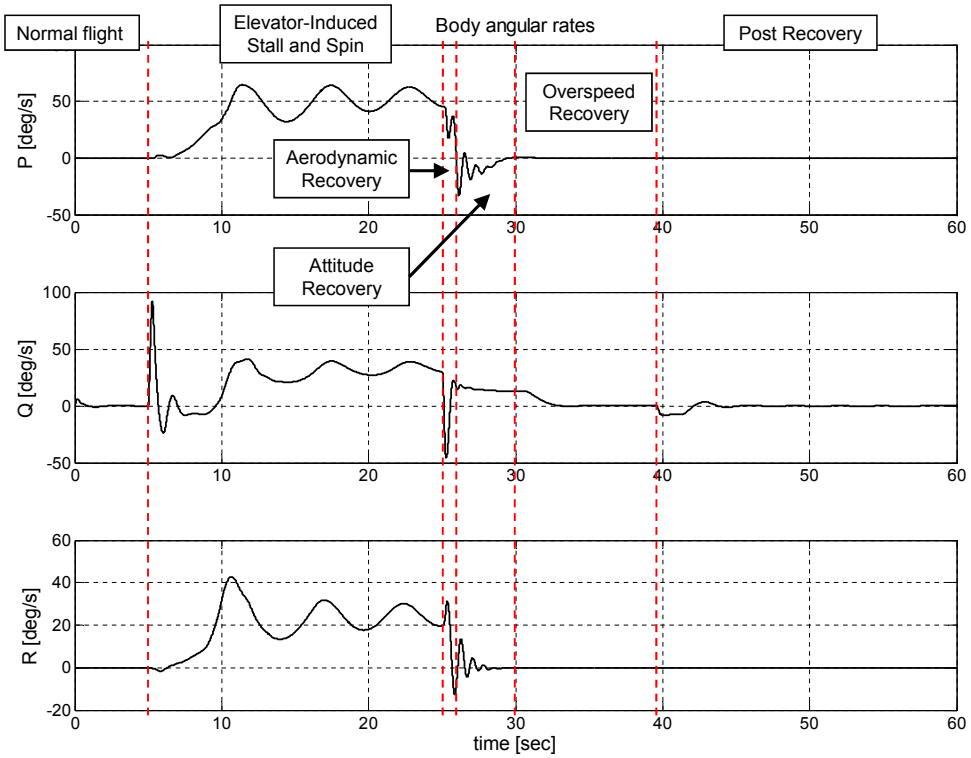


Figure 5.11: State machine based stall and spin recovery: body angular rates vs. time

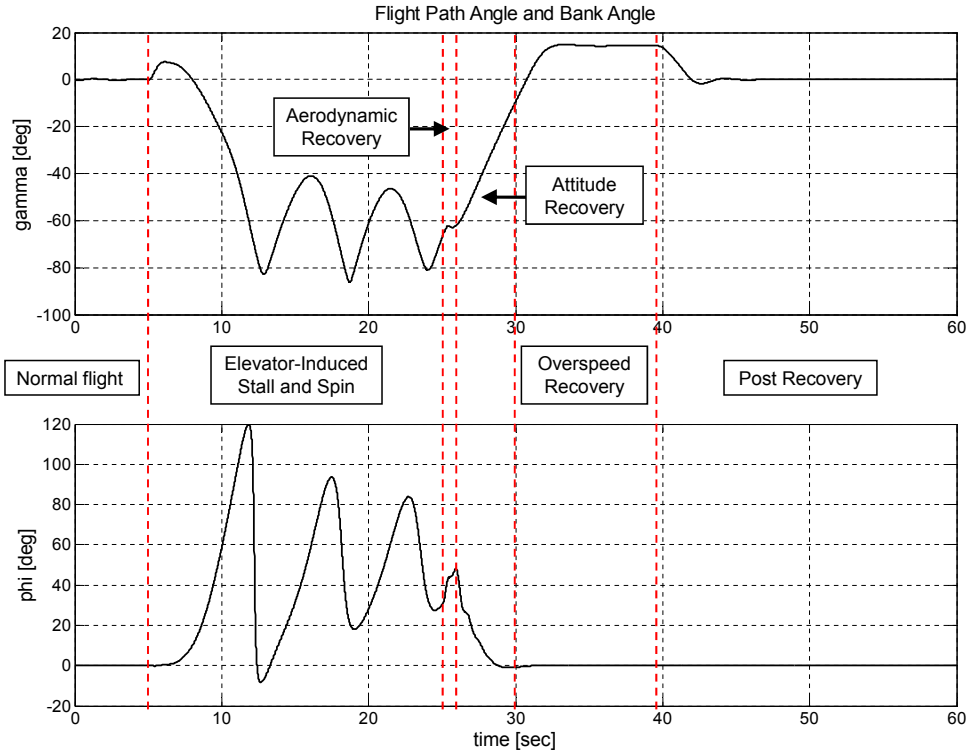


Figure 5.12: State machine based stall and spin recovery: flight path angle and bank angle vs. time

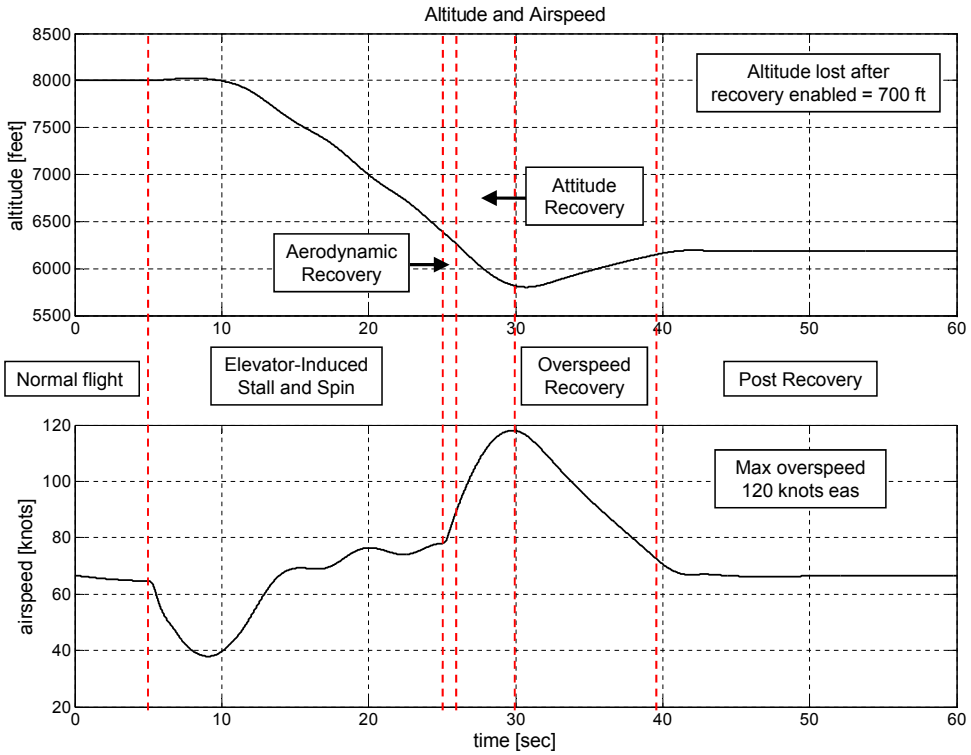


Figure 5.13: State machine based stall and spin recovery: altitude and airspeed vs. time

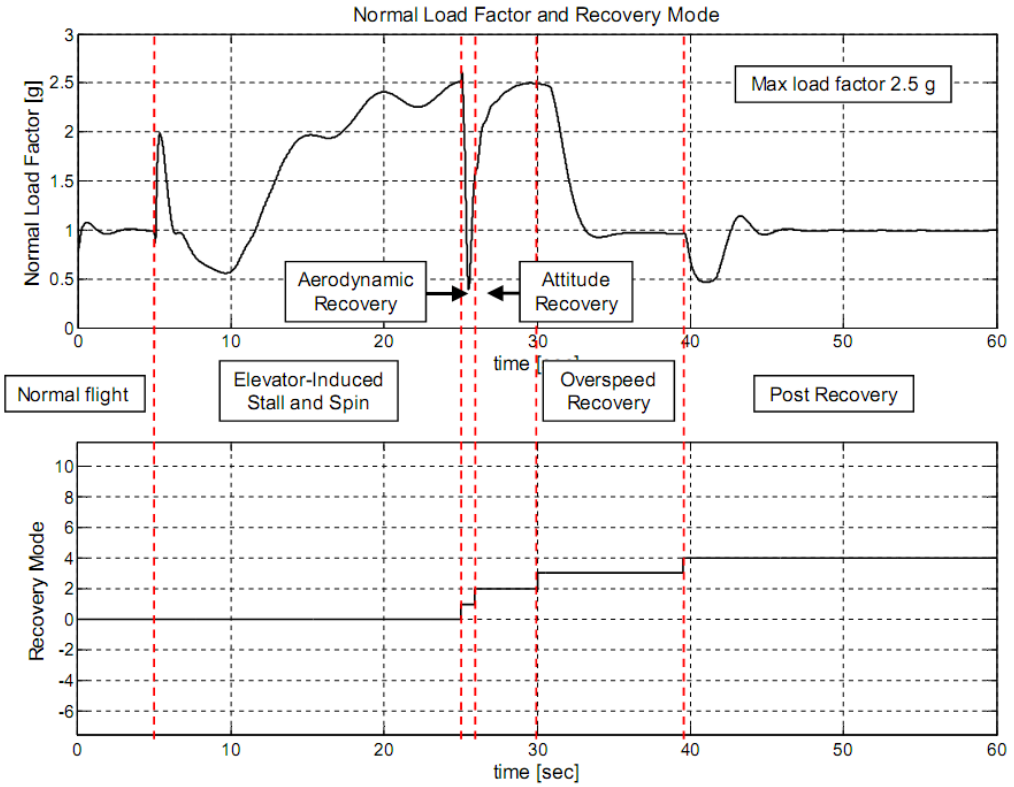


Figure 5.14: State machine based stall and spin recovery: normal load factor and recovery mode vs. time

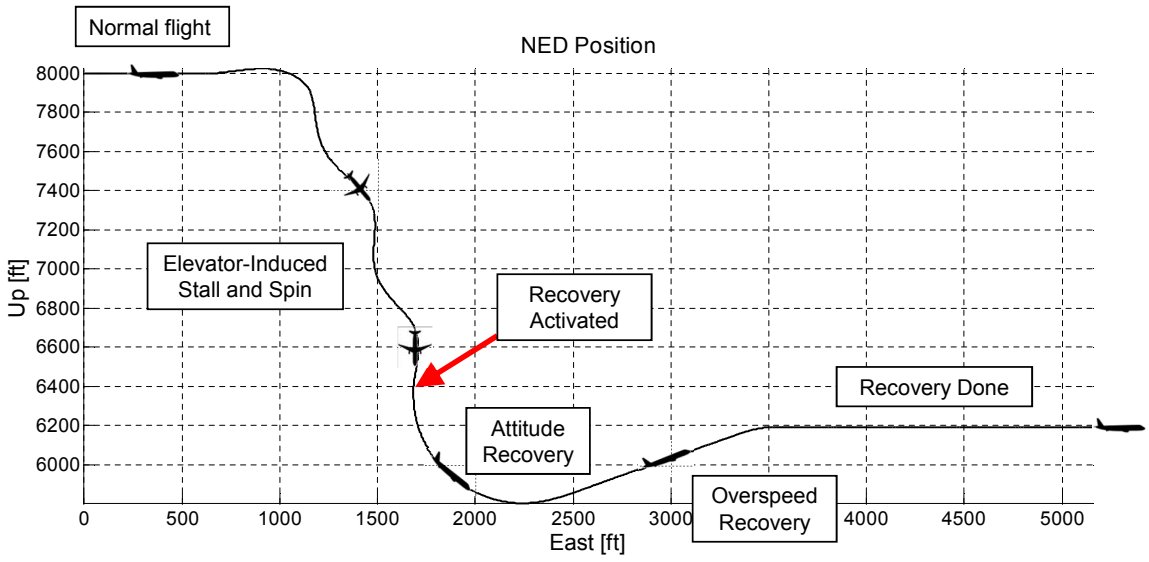


Figure 5.15: State machine based stall and spin recovery: aircraft flight trajectory

5.5 Conclusions

The simulation results show that the state machine based flight envelope recovery system is able to recover the aircraft to normal flight from a severe stall and spin upset condition, given sufficient remaining altitude. During the entire flight envelope recovery procedure, the normal load factor remains within safe limits for structural integrity and passenger safety.

5.6 Summary of Contributions

The following contributions were made in this chapter:

- A state machine based approach for flight envelope recovery was proposed that recovers the flight envelope in stages and uses the passive aerodynamic envelope recovery as the first stage of the recovery.
- A conventional flight control architecture similar to the architecture of fly-by-wire flight control systems used on modern commercial passenger aircraft was implemented for the NASA GTM, and conventional flight control laws and protection functions were designed based on its flight dynamics.
- A state machine was designed to sequentially recover the aerodynamic envelope, the attitude envelope, the flight path angle, and the airspeed.
- The state machine based approach illustrates that once the aerodynamic envelope has been recovered, the conventional flight control laws and protection functions can be used to recover the attitude, the flight path angle, and the airspeed.
- The state machine based flight envelope recovery was implemented on the NASA GTM and verified in simulation. The simulation results showed that the state machine and conventional flight control laws are able to recover the aerodynamic envelope, the attitude envelope, the flight path angle, and the airspeed of the aircraft, and that the normal load factor remains within the structural integrity envelope during the entire flight envelope recovery process.

Chapter 6

Active Aerodynamic Envelope Recovery

In chapter 4, a *passive* aerodynamic recovery method was employed that used the natural rate damping of the aircraft to recover from high angular rates, and used the natural stability of the aircraft's rotational dynamics to recover the angle of attack and the sideslip angle. In this chapter, an *active* aerodynamic envelope recovery method is proposed that employs a Lyapunov-based inner-loop controller to actively recover the angular rates, the angle of attack, and the sideslip angle from out-of-envelope conditions using the control surfaces of the aircraft.

A novel control Lyapunov function is formulated and a set of Lyapunov-based control laws are derived. The resulting Lyapunov-based inner-loop controller not only recovers the angle of attack and sideslip angle, but allows them to be actively controlled in an extended aerodynamic envelope. A further modification allows the controller to also actively control the wind-axis roll rate of the aircraft, in addition to the angle of attack and the sideslip angle. In chapter 7, this ability will be used when recovering the attitude, flight path angle and airspeed of the aircraft.

The Lyapunov-based inner-loop controller is verified in simulation using the NASA Generic Transport Model.

6.1 Problem Formulation

The active aerodynamic envelope recovery is conceptualised as the problem of controlling the angle of attack, sideslip angle and angular rates of the aircraft to recover them from out-of-envelope states, and to regulate them to desired equilibrium states inside the normal aerodynamic envelope. We argue that the problem can be reduced to controlling the rotational dynamics of the aircraft by exploiting the time scale separation between the rigid body rotational dynamics and the point mass translational dynamics of the aircraft. Stated differently, the rotational motion of the aircraft body relative to its own velocity vector, under the influence of the aerodynamic moments, can be controlled over much shorter time scales than the rotational motion of the aircraft velocity vector relative to the inertial reference frame, under the influence of the aerodynamic forces. Over the time scales of the aerodynamic envelope recovery, the aircraft velocity vector may be treated as a slowly varying parameter. The objective of active aerodynamic envelope recovery is therefore to reduce the angular rates of the aircraft and to recover and maintain the attitude of the body axes of the aircraft relative to the wind axes. In simple terms, the objective is to stop the aircraft from spinning and to keep the nose of the aircraft pointing close to the direction of the aircraft airspeed velocity vector.

It should be pointed out that the aerodynamic envelope recovery is not concerned with recovering the gross attitude (pitch angle and bank angle) of the aircraft relative to the inertial reference frame, nor is it concerned with recoverying the flight vector (flight path angle and

airspeed) of the aircraft. The attitude, flight path angle and airspeed recovery shall typically be the responsibility of an outer-loop controller that is designed to control the point mass translational dynamics of the aircraft by commanding the angle of attack, roll rate and engine thrust. (An outer-loop attitude, flight path angle and airspeed recovery controller based on an optimal control formulation will be presented in chapter 7.)

The purpose of aerodynamic envelope recovery is rather to recover the angle of attack, sideslip angle and angular rates of the aircraft to the normal aerodynamic envelope where the aerodynamic forces and moments behave linearly and where the normal flight control laws can be used to recover the gross attitude and point mass translation motion.

6.2 Overview of Lyapunov-Based Controller

A novel Lyapunov-based controller was developed to actively control the angle of attack, sideslip angle, and wind-axis roll rate of the aircraft using the ailerons, elevators and rudder. The Lyapunov-based controller was originally intended only for active aerodynamic envelope recovery, but during the development it was realised that the Lyapunov control law could be modified to accept reference inputs to enable command tracking for the angle of attack, sideslip angle and wind-axis roll rate in an extended aerodynamic envelope. This section provides an overview of the resulting Lyapunov-based controller.

6.2.1 Controller Requirements

The following requirements are formulated for the Lyapunov-based controller:

1. The controller shall be able to actively recover the angle of attack, sideslip angle, and angular rates of the aircraft from out-of-envelope conditions, and shall regulate them towards desired equilibrium states inside the normal aerodynamic envelope.

2. The controller shall be able to actively control the angle of attack, sideslip angle, and wind-axis roll rate of the aircraft to perform command tracking in an extended aerodynamic envelope.

3. The controller shall use feedback from anemometric sensors and inertial sensors similar to those that are typically available on a large transport aircraft.

4. The controller may use the wide-envelope aerodynamic model of the aircraft.

5. The controller is constrained by the actuator limits of the aircraft's aerodynamic control surfaces. Specifically, the achievable steady-state equilibrium states and the achievable speeds of response, will be constrained by the physical and aerodynamic limits of the actuators.

The advantage of the Lyapunov-based controller is that it uses active feedback control to perform the aerodynamic envelope recovery, instead of relying on the natural stability of the aircraft. It therefore has the usual advantages of feedback control, which include stability augmentation, improving the transient response, command tracking, disturbance rejection, and robustness to parameter variations. The disadvantages of the Lyapunov controller are mainly that it relies on the availability of a nonlinear model of the aircraft's flight dynamics, as well as the availability of anemometric sensors that are able to measure the angle of attack and sideslip angle in out-of-envelope conditions. However, it should be noted that the anemometric sensors typically available on commercial passenger airliners have surprisingly wide sensor ranges, and can for example measure angles of attack from -35 to +85 degrees.

6.2.2 Controller Overview and Architecture

The architecture of the proposed Lyapunov-based controller is shown in Figure 6.1. The controller consists of three main components:

- the **Lyapunov Control Law**, that forms the heart of the controller,
- the **Solve Actuator Commands** component, that calculates the required control surface deflections to produce the angular accelerations commanded by the Lyapunov control law,
- and the **Angular Rate References** component, that generates the angular rate references for the Lyapunov control law to help the body axis system track the wind axis system.

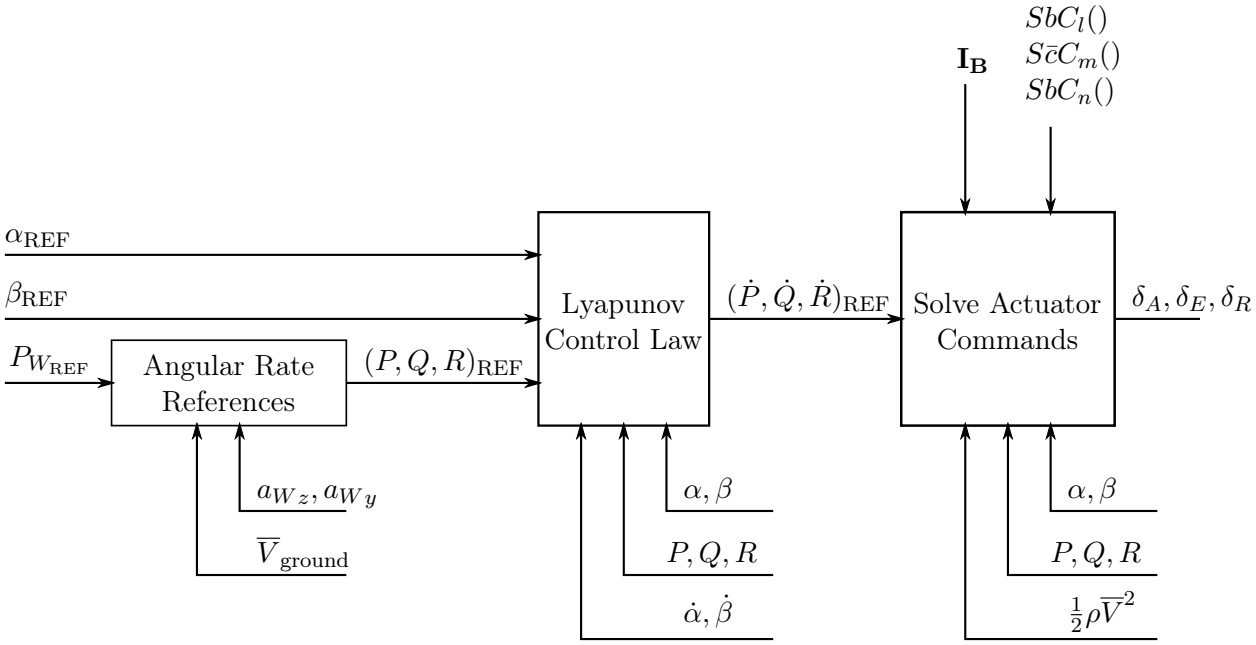


Figure 6.1: Lyapunov-based controller architecture

The Lyapunov-based controller receives an angle of attack reference α_{ref}, a sideslip angle reference β_{ref}, and a wind-axis roll rate reference $P_{W_{\text{ref}}}$ from the pilot or from an outer-loop controller.

The Angular Rate References component provides the body angular rate references P_{ref}, Q_{ref}, and R_{ref} to be tracked by the Lyapunov Control Law. The body angular rate references are obtained from the wind-axis roll rate command $P_{W_{\text{ref}}}$, the natural pitch rate Q_W of the wind axes relative to the inertial axes due to the translational normal acceleration a_{Wz} of the aircraft, and the natural yaw rate R_W of the wind axes relative to inertial axes due to the translational lateral acceleration a_{Wy} of the aircraft. The wind-axis pitch rate and wind-axis yaw rate are indirectly sensed using the translational normal and lateral accelerations as well as the ground speed $\overline{V}_{\text{ground}}$ obtained from the inertial measurement system.

The Lyapunov Control Law calculates the body angular acceleration references $\dot{P}_{\text{ref}}$, $\dot{Q}_{\text{ref}}$, and $\dot{R}_{\text{ref}}$ to make the angle of attack α, the sideslip angle β, and the body angular rates P, Q, and R track their respective references. The Lyapunov Control Law uses feedback from the angle of attack sensor, the sideslip angle sensor, and inertial angular rate sensors. It also uses the time derivatives of the angle of attack and sideslip angle $\dot{\alpha}$ and $\dot{\beta}$ which it obtains by differentiating and filtering the angle of attack and sideslip angle sensor measurements. The

Lyapunov Control Law represents the rotational state of the aircraft with an energy-like function $V(\mathbf{x})$ and calculates the angular accelerations required to make the rate of energy dissipation proportional to the energy itself $\dot{V}(\mathbf{x}) = -KV(\mathbf{x})$, thus ensuring that the rotational state tends towards the reference state with an exponential response.

The Solve Actuator Commands component calculates the required aileron, elevator and rudder deflections (δ_A, δ_E, δ_R) to produce the angular accelerations commanded by the Lyapunov Control Law. Given the required angular accelerations ($\dot{P}_{\text{ref}}$, $\dot{Q}_{\text{ref}}$, $\dot{R}_{\text{ref}}$), the aerodynamic pitching, rolling, and yawing moments (L^A, M^A, N^A) to be applied to the aircraft body are calculated using Euler's equation for rigid body rotational dynamics, with the measured body angular rates (P, Q, R) as known state variables, and the moment of inertia of the aircraft $\mathbf{I_B}$ as a known parameter. Given the required aerodynamic moments (L^A, M^A, N^A), the control surface deflections (δ_A, δ_E, δ_R) are obtained by simultaneously solving the nonlinear equations of the wide-envelope aerodynamic model, with the dynamic pressure $\frac{1}{2}\rho\overline{V}^2$, the angle of attack α, the sideslip angle β, and the body angular rates (P, Q, R) as known state variables, the wide-envelope aerodynamic coefficient functions $C_l()$, $C_m()$, and $C_n()$ for the aerodynamic rolling, pitching, and yawing moments as known functions, and the wing surface area S, the mean aerodynamic chord $\bar{c}$ and the wing span b as known aerodynamic parameters. A multi-variable Secant method is used to iteratively solve the nonlinear equations, and the actuator limits are taken into account.

Since the dynamic pressure $\frac{1}{2}\rho\overline{V}^2$ is used when solving for the control surface deflections, the Lyapunov-based controller has implicit gain scheduling for different airspeeds and altitudes.

6.3 Rigid Body Rotational Dynamics

The plant to be controlled by the Lyapunov controller is the rigid body rotational dynamics of the aircraft. The rotational equations of motion can be grouped into the rotational kinematics and the rotational dynamics. Kinematics is the branch of classical mechanics that studies the motion of a body without considering the causes of the motion, also called the "geometery of the motion", while the dynamics is the branch of classical mechanics that studies the effects of forces and moments on the motion of the body. The rotational kinematics that describes the motion of the body axis system relative to the wind axis system is given by the following equation (as derived by Peddle [5])

$$\begin{bmatrix} \dot{\alpha} \\ \dot{\beta} \\ P_W \end{bmatrix} = \begin{bmatrix} -\cos\alpha\tan\beta & 1 & -\sin\alpha\tan\beta \\ \sin\alpha & 0 & -\cos\alpha \\ \cos\alpha\sec\beta & 0 & \sin\alpha\sec\beta \end{bmatrix} \begin{bmatrix} P \\ Q \\ R \end{bmatrix} + 1/\overline{V} \begin{bmatrix} \sec\beta & 0 \\ 0 & 1 \\ -\tan\beta & 0 \end{bmatrix} \begin{bmatrix} a_{Wz} \\ a_{Wy} \end{bmatrix} \tag{6.1}$$

where α and β are the angle of attack and sideslip angle, P, Q and R are the roll rate, pitch rate and yaw rate coordinated in the body axis system, $\overline{V}$ is the translational velocity, a_{Wy} and a_{Wz} are the total lateral and normal translational accelerations coordinated in wind axes, and P_W is the roll rate of the wind axis system with respect to the inertial axis system. The first term represents the contribution of the aircraft's angular rate relative to the inertial axis system, while the second term represents the contribution of the rotational motion of the translational velocity vector relative to the inertial axis system. The first term therefore represents the fast rotational dynamics, while the second term represents the slow point mass kinematics.

The rotational dynamics that relates the angular acceleration of the aircraft to the moments

acting on the aircraft body is given by the following equation

$$\begin{bmatrix} \dot{P} \\ \dot{Q} \\ \dot{R} \end{bmatrix} = \mathbf{I_B}^{-1} \left(- \begin{bmatrix} P \\ Q \\ R \end{bmatrix} \times \mathbf{I_B} \begin{bmatrix} P \\ Q \\ R \end{bmatrix} + \begin{bmatrix} L \\ M \\ N \end{bmatrix} \right) \tag{6.2}$$

where $\mathbf{I_B}$ is the moment of inertia matrix of the aircraft body, and L, M, and N are the total applied moments. The total moments acting on the aircraft are the sum of the moments due to aerodynamics, thrust, and gravity, as follows

$$L = L_A + L_E + L_G \tag{6.3}$$
$$M = M_A + M_E + M_G \tag{6.4}$$
$$N = N_A + N_E + N_G \tag{6.5}$$

where the subscripts A, E, and G denote the aerodynamic, engine, and gravity moment components respectively. The aerodynamic moments are modelled by the following nonlinear equations

$$L_A = \frac{1}{2}\rho \overline{V}^2 S b C_l(\alpha, \beta, P, Q, R, \delta_A, \delta_E, \delta_R) \tag{6.6}$$
$$M_A = \frac{1}{2}\rho \overline{V}^2 S \bar{c} C_m(\alpha, \beta, P, Q, R, \delta_A, \delta_E, \delta_R) \tag{6.7}$$
$$N_A = \frac{1}{2}\rho \overline{V}^2 S b C_n(\alpha, \beta, P, Q, R, \delta_A, \delta_E, \delta_R) \tag{6.8}$$

where ρ is the air density, S, $\bar{c}$, and b are the surface area, mean aerodynamic chord, and span of the wings, C_l, C_m, and C_n are the aerodynamic coefficient functions for rolling moment, pitch moment, and yawing moment, and δ_A, δ_E, and δ_E are the aileron, elevator and rudder deflections.

The engine moments are the result of the engine thrust vectors not acting through the aircraft centre of mass, as well as the gyroscopic torques associated with the angular momentums of the two engines. The dominant effect of the engines on the rotational dynamics is to produce a pitching moment that is proportional to the total engine thrust. The rolling and yawing moments due to the thrust vectors not acting through the centre of mass tend to oppose each other and cancel out, due to the symmetry of the aircraft and the fact that the left and right engines are normally operated to produce equal thrust. The angular momentums of the engines also tend to oppose each other and cancel out, due to the fact that the engines are designed to rotate in opposite directions and are typically operated at equal engine speeds.

$$L_E \approx 0 \quad M_E = f_M^E(T) \quad N_E \approx 0 \tag{6.9}$$

The gravitational force produces no moments on the aircraft. The gravitational force acts through the centre of mass, and the gravity gradient moment acting on the aircraft is negligible compared to the aerodynamic and engine moments.

$$L_G = 0 \quad M_G = 0 \quad N_G = 0 \tag{6.10}$$

The Lyapunov controller will therefore be designed for the rigid body rotational dynamics, with the rotational motion of the wind axis system relative to the inertial axis system treated as a reference to be tracked, and the pitching moment produced by the engine thrust as a disturbance to be rejected.

6.4 Lyapunov Control Theory

6.4.1 Introduction

This section provides a brief overview of the Lyapunov control theory relevant to the design of the Lyapunov-based inner-loop controller. The textbook on *Applied Nonlinear Control* by Slotine and Li [61] is the primary source of the theory.

When dealing with nonlinear dynamics systems, the familiar tools from linear control theory are less applicable. Linear stability analysis tools can still be used to analyse the local stability of a nonlinear system for small signal behaviour near the linearisation point, but they are no longer reliable when analysing the wide sense stability for large signal behaviour where the system state may move far from the linearisation point. Similarly, linear controllers may still perform well for small range operation where the linear model is still a good approximation of the nonlinear system, but may perform poorly or become unstable for large range operation. We therefore turn to nonlinear stability analysis and control design tools.

The most general approach to studying the stability of nonlinear systems is Lyapunov stability theory. Lyapunov introduced two methods for stability analysis, namely the *linearisation method* and the *direct method*. The linearisation method draws conclusions about a nonlinear system's local stability about an equilibrium point from the stability properties of its linear approximation. The direct method is not restricted to local stability, and determines the stability properties of a nonlinear system by constructing a scalar energy-like function for the system and examining the function's time variation.

Lyapunov's linearisation method is a formalisation of the intuitive concept that a nonlinear system should behave similarly to its linearised approximation for small ranges about the linearisation point. In fact, Lyapunov's linearisation method serves as the fundamental justification for using linear control techniques.

Lyapunov's direct method is the mathematical extension of the intuitive concept that if the total energy of a system is continuously dissipated, then the system, whether linear or nonlinear, must eventually settle to an equilibrium point. The stability of a system may therefore be determined by examining the variation of a single scalar energy-like function, called a Lyapunov function.

6.4.2 Concepts of Stability for Nonlinear Systems

The concept of stability for nonlinear systems is more complex than for linear systems, and a number of more refined stability concepts are defined, namely Lyapunov stability, asymptotic stability, and exponential stability. We also define the domain of attraction of an equilibrium state.

Lyapunov stability means that the system trajectory can be kept arbitrarily close to an equilibrium state by starting sufficiently close to it. If a system is Lyapunov stable, the system trajectory does not necessarily converge to the equilibrium state, but will stay within a bounded domain around the equilibrium state. In many control system applications, Lyapunov stability is not sufficient, and we wish the system trajectory to eventually converge to the equilibrium state.

Asymptotic stability means that the equilibrium state is Lyapunov stable, and in addition, the system trajectory converges to the equilibrium state as time goes to infinity. For many control systems applications, it is still not sufficient to know that a system will converge to the equilibrium point as time tends to infinity, and there is a need to estimate how fast the system trajectory will converge to the equilibrium state. The concept of exponential stability is useful for this purpose.

Exponential stability means that the system trajectory converges to the equilibrium state faster than an exponential function. For asymptotic and exponential stability, only system

trajectories that start sufficiently close to the equilibrium state will converge to the equilibrium state.

The *domain of attraction* of the equilibrium point is defined as the largest set of states from which system trajectories that are initiated at these states will converge to the equilibrium state.

6.4.3 Lyapunov Stability Analysis

Lyapunov stability analysis using Lyapunov's direct method is the mathematical extension of the physical observation that if the total energy of a system is continuously dissipated, then the system, whether linear or nonlinear, must eventually settle to an equilibrium point. The idea is to propose an energy-like candidate function that expresses the system state vector as a single scalar "energy", and then to show that the time derivative of the energy function is always negative or zero in some domain about the equilibrium state. The energy-like function must be zero at the equilibrium state, and must be positive definite for all other states in the domain of interest. If the time derivative of the energy is always negative or zero over the domain, then the equilibrium state is Lyapunov stable. If the time derivative of the energy is always strictly negative over the domain, then the equilibrium is asymptotically stable, and trajectories starting in the domain will eventually converge to the equilibrium state as time goes to infinity. If the time derivative of the energy is always strictly negative and also proportional to the energy over the domain, then the equilibrium is exponentially stable, and trajectories starting in the domain will converge to the equilibrium state faster than an exponential function. These concepts of Lyapunov stability can be expressed mathematically as follows:

Lyapunov Stability

Given an autonomous nonlinear dynamic system

$$\dot{\mathbf{x}}(t) \quad = \quad \mathbf{f}(\mathbf{x}(t)) \tag{6.11}$$

If a scalar function $V(\mathbf{x})$ can be found that is positive definite and has continuous partial derivatives in a domain $\mathbf{\Omega}$, and if its time derivative $\dot{V}(\mathbf{x})$ along any state trajectory is negative semi-definite, i.e.

- $V(\mathbf{x}) = 0$ when $\mathbf{x} = \mathbf{0}$
- $V(\mathbf{x}) > 0$ when $\mathbf{x} \neq \mathbf{0}$
- $\dot{V}(\mathbf{x}) \leq 0$

then $V(\mathbf{x})$ is a Lyapunov function for the system, the equilibrium state $\mathbf{x} = \mathbf{0}$ is Lyapunov stable, and the domain $\mathbf{\Omega}$ is a domain of attraction of the equilibrium.

Asymptotic Stability

If the time derivative of the Lyapunov function $\dot{V}(\mathbf{x})$ is negative definite, i.e.

- $\dot{V}(\mathbf{x}) < 0$

then the equilibrium state $\mathbf{x} = \mathbf{0}$ is asymptotically stable.

Exponential Stability

If the time derivative of the Lyapunov function $\dot{V}(\mathbf{x})$ is negative definite and proportional to the Lyapunov function, i.e.

- $\dot{V}(\mathbf{x}) = -KV(\mathbf{x})$ with $K > 0$

then the equilibrium state $\mathbf{x} = \mathbf{0}$ is exponentially stable.

Finding a Lyapunov Function

The main drawback of Lyapunov's direct method is the difficulty of finding a Lyapunov function for a given nonlinear system. There is no general method for finding Lyapunov functions for nonlinear systems, and one therefore has to use experience, intuition, and physical insights to search for an appropriate Lyapunov function. However, there are no guarantees, and very often an appropriate Lyapunov function cannot be found for a given system.

6.4.4 La Salle's Invariant Set Theorem

La Salle's invariant set theorem allows us to extend the basic Lyapunov stability analysis to conclude asymptotic or exponential stability in cases where

- The Lyapunov function is not strictly positive definite, but only positive semi-definite, i.e. $V(\mathbf{x}) \geq 0$ for $\mathbf{x} \neq \mathbf{0}$
- The time derivative of the Lyapunov function is not strictly negative definite, but only negative semi-definite $\dot{V}(\mathbf{x}) \leq 0$ for $\mathbf{x} \neq \mathbf{0}$

If a scalar function $V(\mathbf{x})$ can be found that has continuous partial derivatives in a domain $\boldsymbol{\Omega}$, and if its time derivative $\dot{V}(\mathbf{x})$ along any state trajectory is negative semi-definite, and if no system trajectories can stay at states where the function's time derivative equals zero i.e.

- $\dot{V}(\mathbf{x}) \leq 0$
- No system trajectories can stay at states where $\dot{V}(\mathbf{x}) = 0$, except for the trivial trajectory $\mathbf{x} = \mathbf{0}$

then the equilibrium state $\mathbf{x} = \mathbf{0}$ is asymptotically stable, and the domain $\boldsymbol{\Omega}$ is a domain of attraction of the equilibrium.

6.4.5 Lyapunov Control Design

In Lyapunov control design, a control law is designed for a given open-loop nonlinear dynamic system to make a given scalar function a Lyapunov function of the closed-loop system. Given the following autonomous nonlinear dynamic system

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) \tag{6.12}$$

where $\mathbf{x}(t)$ is the state vector and $\mathbf{u}(t)$ is the input vector, the objective is to find a control law

$$\mathbf{u} = \mathbf{g}(\mathbf{x}) \tag{6.13}$$

where the control law $\mathbf{g}$ is a nonlinear vector function that makes a scalar function $V(\mathbf{x})$ a Lyapunov function of the closed-loop system

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}, \mathbf{u}) \tag{6.14}$$

$$= \mathbf{f}(\mathbf{x}, \mathbf{g}(\mathbf{x}) \tag{6.15}$$

$$= \mathbf{f}(\mathbf{x}) \tag{6.16}$$

There are basically two approaches to using Lyapunov stability theory for control design, and both use trial and error. The first approach involves hypothesizing the control law and then finding a Lyapunov function to justify the choice. The second approach involves hypothesizing a Lyapunov function candidate and then finding a control law to make the candidate a Lyapunov function.

In general, the application of Lyapunov stability theory to control design is more easily rewarding than Lyapunov stability analysis, since one often has the freedom to modify the dynamics of the nonlinear dynamic system, by designing the controller in such a way that a chosen scalar function becomes a Lyapunov function for the closed-loop system.

6.5 Lyapunov Controller Design

In this section, the detailed design of the Lyapunov-based controller is presented. First, the design of the Lyapunov Control Laws is presented. A novel candidate Lyapunov function is proposed and the feedback control laws that make the candidate function a Lyapunov function are derived. Practical guidelines for choosing the Lyapunov control gains are provided. Second, the design of the Solve Actuator Commands component is presented. The required aerodynamic moments are calculated using Euler's equation for rigid body rotational dynamics, and then the required control surface deflections are determined by simultaneously solving the nonlinear equations of the wide-envelope aerodynamic model. The Solve Actuator Commands component provides implicit gain scheduling for the Lyapunov-based controller, and also accommodates actuator saturation. When the control surfaces saturate, the Lyapunov controller may no longer be able to provide exponential stability, but still attempts to provide asymptotic stability. Third, the design of the Angular Rate References component is presented. The component combines the wind-axis roll rate reference provided by the pilot or outer-loop control law, with the wind-axis pitch rate and yaw rate due to the translational motion of the aircraft to generate body angular rate references for the Lyapunov Control Law. The wind-axis pitch rate and yaw rate are sensed in a novel way using the normal and lateral specific accelerations, and the gross aircraft attitude provided by the inertial sensors. The steady-state command tracking performance of the Lyapunov-based controller is improved by adding integral terms. Limited integrators are used so that the steady-state tracking is improved without affecting the stability or transient response. Finally, the robustness of the Lyapunov-based controller to model uncertainty is considered.

6.5.1 Lyapunov Control Laws

The design of a Lyapunov control law is not a straightforward task. One does not simply design a Lyapunov controller. The key challenge is the difficulty of finding an appropriate candidate Lyapunov function for which a control law can be derived to make the candidate a Lyapunov function. Since there is no general method for finding Lyapunov functions for nonlinear systems, a trial and error approach has to be used, and the designer has to rely on experience, intuition, and physical insights. However, there are no guarantees, and it can take very long to find an appropriate Lyapunov function, if one can be found at all.

Fortunately, after many trials and errors with several unsuccessful candidate Lyapunov functions, we were eventually able to propose a novel candidate Lyapunov function for which a set of control laws could successfully be derived to make the candidate function a Lyapunov function. The candidate Lyapunov function and the derivation of the Lyapunov control laws are presented below.

We define the state vector to be

$$\mathbf{x} = \begin{bmatrix} (\alpha - \alpha_{\text{ref}}) & (\beta - \beta_{\text{ref}}) & (P - P_B^{WI}) & (Q - Q_B^{WI}) & (R - R_B^{WI}) \end{bmatrix}^T \tag{6.17}$$

where α_{ref} and β_{ref} are the angle of attack and sideslip angles references, and P_B^{WI}, Q_B^{WI}, and R_B^{WI} are the angular rate components of the wind axis system relative to the inertial axis system, coordinated in the body axis system. This definition of the state vector represents the fact that we want the angle of attack and sideslip angle to follow the reference commands, while also matching the angular rate of the body axis with the angular rate of the wind axis system.

We then propose the following novel Lyapunov function candidate

$$V(\mathbf{x}) = \frac{1}{2}r(\mathbf{x})^2 \tag{6.18}$$

with

$$\begin{aligned} r(\mathbf{x}) = {} & k_\alpha(\alpha - \alpha_{\text{ref}}) + k_\beta(\beta - \beta_{\text{ref}}) + \\ & k_P(P - P_B^{WI}) + k_Q(Q - Q_B^{WI}) + k_R(R - R_B^{WI}) \end{aligned} \tag{6.19}$$

The Lyapunov function candidate represents the deviation of body axis system from the wind axis system as an energy function. Note that this energy function is not a true Lyapunov function candidate, because it is not positive definite, but only positive semi-definite. The energy function is not only zero at the desired state, but is also zero on the entire hyperplane defined by

$$k_\alpha(\alpha - \alpha_{\text{ref}}) + k_\beta(\beta - \beta_{\text{ref}}) + k_P(P - P_B^{WI}) + k_Q(Q - Q_B^{WI}) + k_R(R - R_B^{WI}) = 0 \tag{6.20}$$

However, if we can show that this hyperplane is not an invariant set of the system, we can use La Salle's Invariant Set Theorem to still infer asymptotic Lyapunov stability, and even exponential stability.

The time derivative of the energy function can then be derived as

$$\dot{V}(\mathbf{x}) = \frac{d}{dt}[\frac{1}{2}r(\mathbf{x})^2] \tag{6.21}$$

$$\dot{V}(\mathbf{x}) = r(\mathbf{x})\dot{r}(\mathbf{x}) \tag{6.22}$$

where the time derivative of $r(\mathbf{x})$ is calculated as

$$\dot{r}(\mathbf{x}) = k_\alpha\dot{\alpha} + k_\beta\dot{\beta} + k_P\dot{P} + k_Q\dot{Q} + k_R\dot{R} \tag{6.23}$$

Substituting $r(\mathbf{x})$ and $\dot{r}(\mathbf{x})$ from equations (6.19) and (6.23) into equation (6.22), we obtain an explicit expression for the Lyapunov time derivative in terms of the states of the system and their time derivatives

$$\begin{aligned} \dot{V}(\mathbf{x}) = {} & [k_\alpha(\alpha - \alpha_{\text{ref}}) + k_\beta(\beta - \beta_{\text{ref}}) + k_P(P - P_B^{WI}) + k_Q(Q - Q_B^{WI}) + k_R(R - R_B^{WI})] \\ & [k_\alpha\dot{\alpha} + k_\beta\dot{\beta} + k_P\dot{P} + k_Q\dot{Q} + k_R\dot{R}] \end{aligned} \tag{6.24}$$

To ensure exponential Lyapunov stability, we wish the time derivative of the Lyapunov function to be negative and proportional to the Lyapunov function itself. This is expressed mathematically as

$$\dot{V}(\mathbf{x}) = -KV(\mathbf{x}) \tag{6.25}$$

where $K > 0$ is a constant scalar value that determines the convergence rate for the exponential Lyapunov stability. Substituting $V(\mathbf{x})$ and $\dot{V}(\mathbf{x})$ from equations (6.18) and (6.22) we obtain

$$r(\mathbf{x})\dot{r}(\mathbf{x}) = -K\frac{1}{2}r(\mathbf{x})^2 \tag{6.26}$$

We can then divide both sides of the equation by $r(\mathbf{x})$, in the domain where $r(\mathbf{x})$ is nonzero, i.e. for all nonzero values of the Lyapunov function

$$\dot{r}(\mathbf{x}) = -\frac{1}{2}Kr(\mathbf{x}) \tag{6.27}$$

We can now substitute $r(\mathbf{x})$ and $\dot{r}(\mathbf{x})$ from equations (6.19) and (6.23) to obtain an explicit differential equation in terms of the states and their time derivatives, that will ensure exponential Lyapunov stability

$$\begin{aligned}&(k_\alpha\dot{\alpha} + k_\beta\dot{\beta} + k_P\dot{P} + k_Q\dot{Q} + k_R\dot{R}) = \\ &\quad -\frac{1}{2}K[k_\alpha(\alpha - \alpha_{\text{ref}}) + k_\beta(\beta - \beta_{\text{ref}}) + k_P(P - P_B^{WI}) + k_Q(Q - Q_B^{WI}) + k_R(R - R_B^{WI})]\end{aligned} \tag{6.28}$$

Next, we can group and equate the terms on the left side of the equation with appropriate terms on the right side of the equation, as follows

$$k_P\dot{P} = -\frac{1}{2}K[k_P(P - P_B^{WI})] \tag{6.29}$$

$$k_\alpha\dot{\alpha} + k_Q\dot{Q} = -\frac{1}{2}K[k_\alpha(\alpha - \alpha_{\text{ref}}) + k_Q(Q - Q_B^{WI})] \tag{6.30}$$

$$k_\beta\dot{\beta} + k_R\dot{R} = -\frac{1}{2}K[k_\beta(\beta - \beta_{\text{ref}}) + k_R(R - R_B^{WI})] \tag{6.31}$$

Finally, we can rearrange the equations to make $\dot{P}$, $\dot{Q}$, and $\dot{R}$ the subjects of the respective equations

$$\dot{P} = -\frac{1}{2}K(P - P_B^{WI}) \tag{6.32}$$

$$\dot{Q} = -\frac{1}{2}K\frac{k_\alpha}{k_Q}(\alpha - \alpha_{\text{ref}}) - \frac{1}{2}K(Q - Q_B^{WI}) - \frac{k_\alpha}{k_Q}\dot{\alpha} \tag{6.33}$$

$$\dot{R} = -\frac{1}{2}K\frac{k_\beta}{k_R}(\beta - \beta_{\text{ref}}) - \frac{1}{2}K(R - R_B^{WI}) - \frac{k_\beta}{k_R}\dot{\beta} \tag{6.34}$$

These equations now represent three control laws that ensure exponential Lyapunov stability, one for the roll axis, one for the pitch axis, and one for the yaw axis. Inspecting the control laws, it is pleasing to see that they make intuitive sense, and look similar to control laws that might have been obtained using linear time-invariant design techniques. The roll axis angular acceleration is made proportional to the roll rate error. The pitch axis angular acceleration is made proportional to the weighted sum of the angle of attack error, the pitch rate error, and the angle of attack rate. The yaw axis angular acceleration is made proportional to the weighted sum of the sideslip angle error, the yaw rate error, and the sideslip angle rate.

Choosing the Lyapunov Control Gains

This section provides some guidelines for tuning the control gains in the Lyapunov-based controller. We consider the gain K with which the time derivative of the Lyapunov function $\dot{V}(\mathbf{x})$ is negatively proportional to the Lyapunov function $V(\mathbf{x})$,

$$\dot{V}(\mathbf{x}) = -KV(\mathbf{x}) \text{ with } K > 0 \tag{6.35}$$

as well as the gains k_α, k_β, k_P, k_Q, and k_R that are used in the Lyapunov function itself

$$V(\mathbf{x}) = \frac{1}{2}r(\mathbf{x})^2 \tag{6.36}$$

with

$$\begin{aligned} r(\mathbf{x}) &= k_\alpha(\alpha - \alpha_{\text{ref}}) + k_\beta(\beta - \beta_{\text{ref}}) + \\ &\quad k_P(P - P_B^{WI}) + k_Q(Q - Q_B^{WI}) + k_R(R - R_B^{WI}) \end{aligned} \tag{6.37}$$

We note that the gain K determines the rate of exponential convergence of the "energy" of the system, and therefore in effect determines the rate of convergence of the "envelope" of the state vector to the desired state.

$$V(\mathbf{x}(t)) = e^{-Kt}V(\mathbf{0}) \tag{6.38}$$

The gain K therefore determines the rate of convergence of the state vector as a whole. On the other hand, the gains k_α, k_β, k_P, k_Q, and k_R determine the relative weighting of the individual states within the state vector. These gains can therefore be tuned to trade off the relative rates of convergence of the individual states within the state vector.

Considering the Lyapunov control laws themselves, we see another perspective on the gains k_α, k_β, k_P, k_Q, and k_R. The weighting gains also act as "spring" and "damper" terms.

$$k_P\dot{P} = -\frac{1}{2}K[k_P(P - P_B^{WI})] \tag{6.39}$$

$$k_\alpha\dot{\alpha} = -\frac{1}{2}K[k_\alpha(\alpha - \alpha_{\text{ref}}) + k_Q(Q - Q_B^{WI})] - k_Q\dot{Q} \tag{6.40}$$

$$k_\beta\dot{\beta} = -\frac{1}{2}K[k_\beta(\beta - \beta_{\text{ref}}) + k_R(R - R_B^{WI})] - k_R\dot{R} \tag{6.41}$$

The gains k_α and k_β produce restoring moments for the angle of attack and sideslip angle respectively, while the gains k_P, k_Q, and k_R provide angular rate damping on pitch rate, roll rate, and yaw rate respectively.

Here are some examples of how we can use these insights to tune the Lyapunov controller gains:

1. If the overall speed of response of the state vector is too slow, then we could increase the rate of convergence of the "energy" by increasing the gain K.

2. If we would like to make the angle of attack regulation more aggressive relative to the sideslip angle regulation, then we could increase the angle of attack restoring gain k_α

3. If the angle of attack transient response is too oscillatory, then we could increase the pitch rate damping gain k_Q.

Satisfying La Salle's Invariant Set Theorem

To satisfy La Salle's Invariant Set Theorem, we need to show that no system trajectories can stay at states where $\dot{V}(\mathbf{x}) = 0$, except for the trivial trajectory $\mathbf{x} = \mathbf{0}$. Note that because the Lyapunov control law is designed to make the Lyapunov energy rate negative and proportional to the Lyapunov energy ($\dot{V}(\mathbf{x}) = -KV(\mathbf{x})$), the only states where the Lyapunov energy rate is zero ($\dot{V}(\mathbf{x}) = 0$) are the states where the Lyapunov energy itself is also zero ($V(\mathbf{x}) = 0$). We therefore check that La Salle's Invariant Set Theorem is satisfied by performing Monte Carlo simulations where state trajectories are simulated from a comprehensive set of initial states uniformly distributed over the hyperplane where $V(\mathbf{x}) = 0$, defined by

$$k_\alpha(\alpha - \alpha_{\text{ref}}) + k_\beta(\beta - \beta_{\text{ref}}) + k_P(P - P_B^{WI}) + k_Q(Q - Q_B^{WI}) + k_R(R - R_B^{WI}) = 0 \quad (6.42)$$

using the differential equations for the reduced-order model of the fast rotational dynamics with the Lyapunov control law added

$$\begin{aligned}
\dot{\alpha} &= -\cos\alpha\tan\beta P + Q - \sin\alpha\tan\beta R && (6.43)\\
\dot{\beta} &= \sin\alpha P - \cos\alpha R && (6.44)\\
\dot{P} &= -\frac{1}{2}K(P - P_B^{WI}) && (6.45)\\
\dot{Q} &= -\frac{1}{2}K\frac{k_\alpha}{k_Q}(\alpha - \alpha_{\text{ref}}) - \frac{1}{2}K(Q - Q_B^{WI}) - \frac{k_\alpha}{k_Q}\dot{\alpha} && (6.46)\\
\dot{R} &= -\frac{1}{2}K\frac{k_\beta}{k_R}(\beta - \beta_{\text{ref}}) - \frac{1}{2}K(R - R_B^{WI}) - \frac{k_\beta}{k_R}\dot{\beta} && (6.47)
\end{aligned}$$

Monte Carlo simulations were performed with the aircraft state randomly initialised with angles of attack uniformly distributed from -5 to +85 degrees and the body angular rates uniformly distributed between -45 and +45 degrees per second. The sideslip angle was then calculated using equation 6.42 to ensure that the initial state is on the hyperplane where $V(\mathbf{x}) = 0$. A set of 1000 Monte Carlo simulations were performed. The time histories of the angle of attack and the sideslip angle are plotted in figure 6.2, the time histories of the body angular rates are plotted in figure 6.3, and the time histories of the Lyapunov energy and the Lyapunov energy rate are plotted in figure 6.4. Scatter plots of the initial angles of attack, initial sideslip angles and initial body angular rates are shown in figure 6.5 to show the coverage of the Monte Carlo simulations. The results shown in these figures were generated for the angle of attack reference set to 4 degrees and the sideslip angle reference set to 0 degrees, but similar results are obtained for other angle of attack and sideslip angle references.

The time histories of the angle of attack, sideslip angle and body angular rates show that all state trajectories converge to the desired equilibrium state corresponding to the given angle of attack reference and sideslip angle reference, and that no system trajectories stay at any other states where $\dot{V}(\mathbf{x}) = 0$. (The time histories of the Lyapunov energy and the Lyapunov derivative are plotted to show that both the Lyapunov energy and the Lyapunov energy rate remain zero for all state trajectories starting on the hyperplane.) This confirms that La Salle's Invariant Set Theorem is satisfied, and that the Lyapunov control law will provide asymptotic stability for states where the Lyapunov energy is zero.

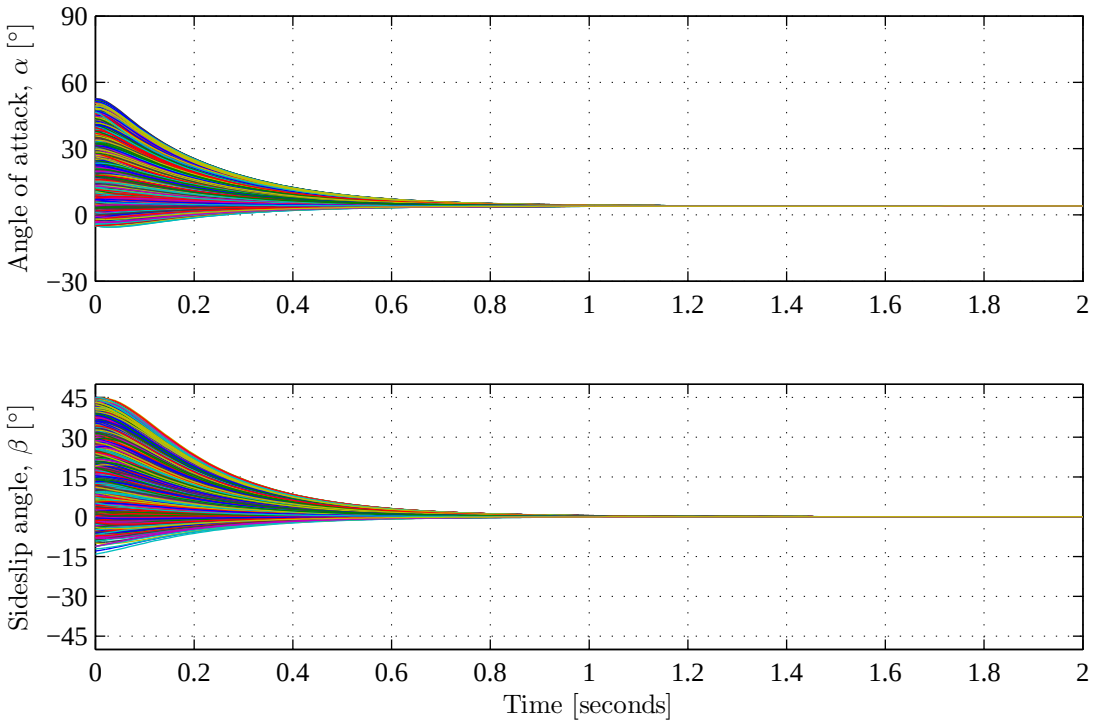


Figure 6.2: Satisfying La Salle's Invariant Set Theorem: time histories of angle of attack and sideslip angle

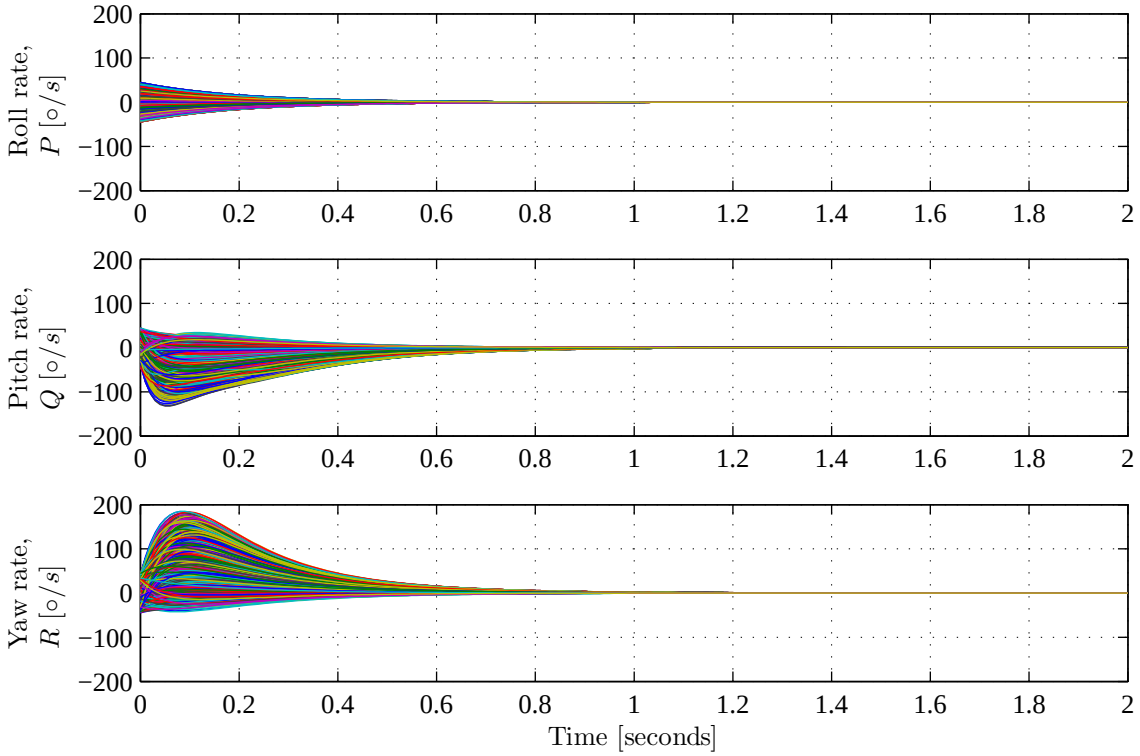


Figure 6.3: Satisfying La Salle's Invariant Set Theorem: time histories of body angular rates

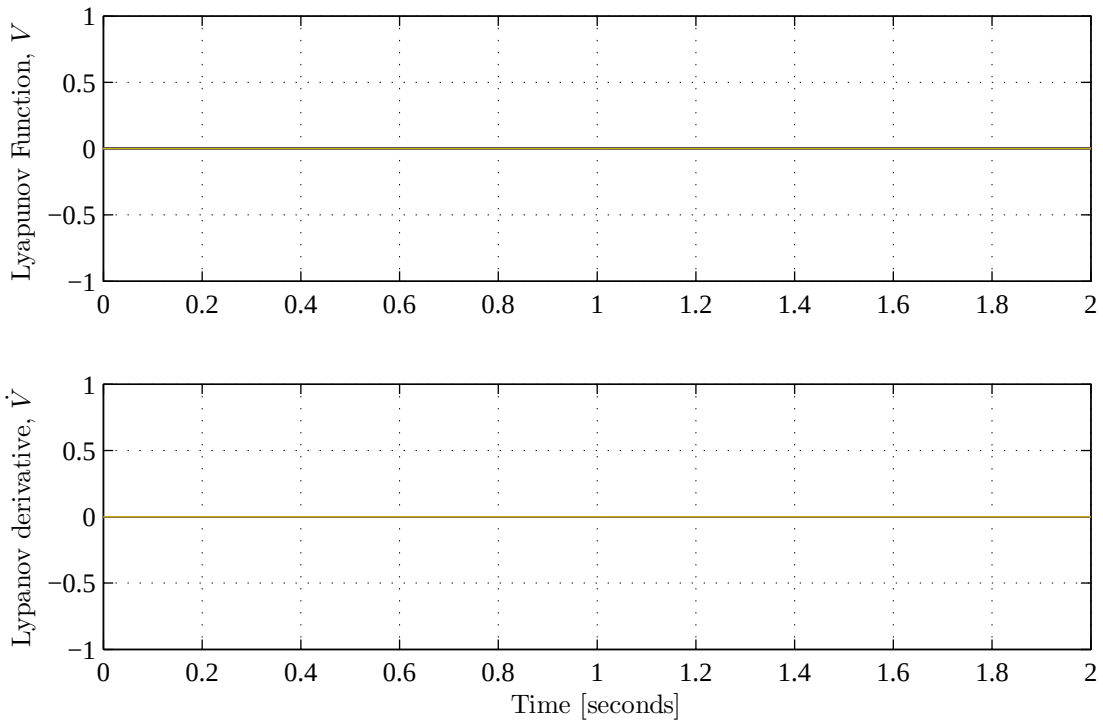


Figure 6.4: Satisfying La Salle's Invariant Set Theorem: time histories of Lyapunov energy and energy rate

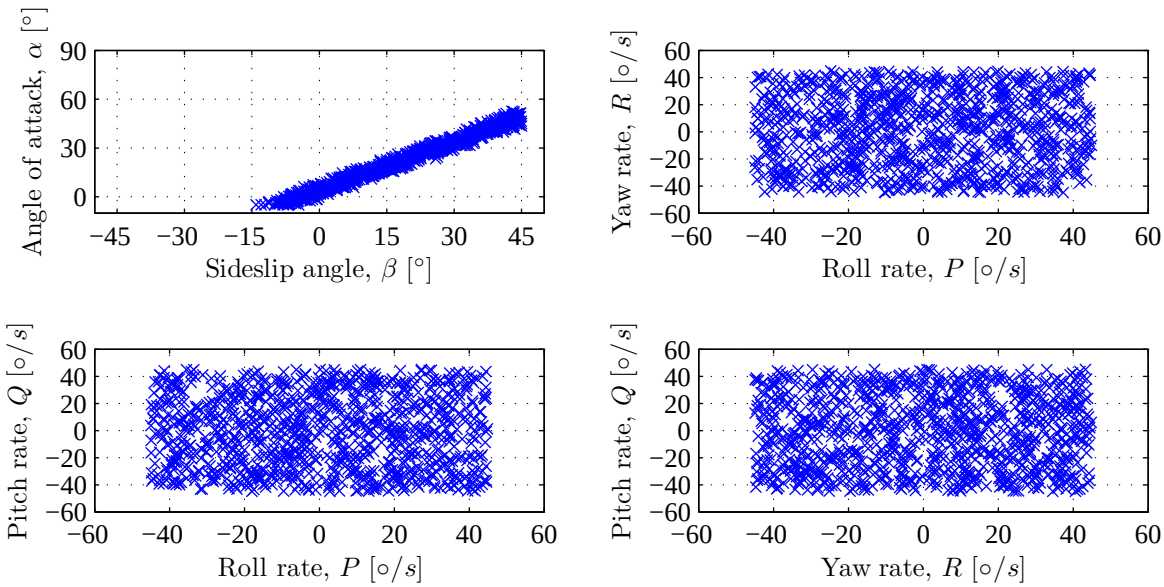


Figure 6.5: Satisfying La Salle's Invariant Set Theorem: scatter plots of initial angle of attack, sideslip angle, and body angular rates

6.5.2 Solving for the Required Control Surface Deflections

Given the angular acceleration commands $\dot{P}_{\text{ref}}$, $\dot{Q}_{\text{ref}}$, and $\dot{R}_{\text{ref}}$ supplied by the Lyapunov control laws to ensure exponential stability, we now wish to calculate the required control surface

deflections δ_A, δ_E, and δ_R for the ailerons, elevator and rudder to produce these angular accelerations. We will first use Euler's well-known equation for rigid body rotational dynamics to calculate the required aerodynamic moments L^A, M^A and N^A, and then we will solve the wide-envelope aerodynamic model to determine the required control surface deflections.

Handling Inertial Coupling

The solution of the actuator deflections implicitly handles inertial coupling by using Euler's equation for rigid body rotational dynamics to calculate the required aerodynamic moments. Euler's equation models the gyroscopic torques that produce the inertial coupling, and the calculated aerodynamic torques therefore already compensate for the gyroscopic torques.

Implicit Gain Scheduling

The solution of the actuator deflections implicitly provides gain scheduling for the Lyapunov controller at different airspeeds and altitudes, due to the fact that the dynamic pressure is taken into account when the wide-envelope aerodynamic model is used to calculate the aerodynamic moments. The implicit gain scheduling is feasible because the airspeed and altitude of the aircraft changes over much longer time scales than the rotational variables, which allows the dynamic pressure to be used as a slowly varying scheduling parameter.

Implications of Actuator Saturation

It should be noted that the angular accelerations commanded by the Lyapunov controller may require aerodynamic moments that are greater than what the control surfaces can actually produce. Furthermore, the aerodynamic moments may in some conditions be dominated by the static moments due to angle of attack and sideslip angle, and the dynamic moments due to the body angular rates, which could dwarf the control moments produced by the ailerons, elevator and rudder. In some cases the static and dynamic moments may aid the control moments, but in other cases the static and dynamic moments may oppose the control moments. The important point is that the Lyapunov control law may require control moments that are greater than what the control surfaces can produce, resulting in saturation of the control surface deflections.

The actuator saturation can be managed to some degree by reducing the Lyapunov control gains and making the controller transient response less agressive, but ultimately the algorithm that solves for the required control surface deflections must take the actuator constraints into account.

When the actuators saturate, the Lyapunov controller can no longer ensure exponential stability, where the energy rate is negatively proportional to the energy. However, even when the actuators saturate, the Lyapunov controller still gives its best effort to provide exponential stability, and is therefore likely to achieve at least asymptotic stability, where the energy rate is at least negative.

Since the Lyapunov time derivative is available from the Lyapunov controller, a supervisory system can monitor the energy rate to check in real time whether asymptotic stability is achieved.

Calculating the Required Aerodynamic Moments

Given the angular acceleration commands, the total moments to be applied to the aircraft body are calculated using the Euler's equation for rigid body rotational dynamics

$$\begin{bmatrix} L_{\text{ref}} \\ M_{\text{ref}} \\ N_{\text{ref}} \end{bmatrix} = \mathbf{I_B} \begin{bmatrix} \dot{P}_{\text{ref}} \\ \dot{Q}_{\text{ref}} \\ \dot{R}_{\text{ref}} \end{bmatrix} + \begin{bmatrix} P \\ Q \\ R \end{bmatrix} \times \mathbf{I_B} \begin{bmatrix} P \\ Q \\ R \end{bmatrix} \tag{6.48}$$

where L_{ref}, M_{ref}, and N_{ref} are the total commanded rolling, pitching and yawing moments, $\dot{P}_{\text{ref}}$, $\dot{Q}_{\text{ref}}$, and $\dot{R}_{\text{ref}}$ are the angular acceleration commands, P, Q, and R are the known body angular rates measured by the inertial sensors, and $\mathbf{I_B}$ is the known moment of inertia of the aircraft.

We then recognise that the total moments acting on the aircraft body are the sum of the aerodynamic moments and the thrust moments,

$$\begin{aligned} L &= L^A && (6.49)\\ M &= M^A + M^E && (6.50)\\ N &= N^A && (6.51) \end{aligned}$$

However, we will treat the pitching moment M^E produced by the engine thrust as a disturbance torque, and we will make required the aerodynamic moments equal to the required total moments.

$$\begin{aligned} L^A_{\text{ref}} &= L_{\text{ref}} && (6.52)\\ M^A_{\text{ref}} &= M_{\text{ref}} && (6.53)\\ N^A_{\text{ref}} &= N_{\text{ref}} && (6.54) \end{aligned}$$

Now that we have isolated the required aerodynamic moments, we can proceed to determine the control surface deflections to produce these moments, using the wide-envelope aerodynamic model. We therefore wish to simultaneously solve the following set of nonlinear equations

$$\begin{aligned} L^A_{\text{ref}} &= \frac{1}{2}\rho\overline{V}^2 SbC_l(\alpha,\beta,P,Q,R,\delta_A,\delta_E,\delta_R) && (6.55)\\ M^A_{\text{ref}} &= \frac{1}{2}\rho\overline{V}^2 S\bar{c}C_m(\alpha,\beta,P,Q,R,\delta_A,\delta_E,\delta_R) && (6.56)\\ N^A_{\text{ref}} &= \frac{1}{2}\rho\overline{V}^2 SbC_n(\alpha,\beta,P,Q,R,\delta_A,\delta_E,\delta_R) && (6.57) \end{aligned}$$

where L^A_{ref}, M^A_{ref}, and N^A_{ref} are the target aerodynamic moments and δ_A, δ_E, and δ_R are the control surface deflections to be solved.

For a given aircraft state, the dynamic pressure $\frac{1}{2}\rho\overline{V}^2$, the angle of attack α, the sideslip angle β, and the body angular rates, P, Q, and R, are all known variables, since they are measured by the anemometric and inertial sensors. The wing surface area S, the mean aerodynamic chord $\bar{c}$ and the wing span b are known aerodynamic parameters, and the functions C_l, C_m and C_n are the known aerodynamic coefficient functions representing the wide-envelope aerodynamic model. The only remaining unknowns to be solved are aileron deflection δ_A, elevator deflection δ_E, and rudder deflection δ_R.

Equations 6.55, 6.56 and 6.57 therefore represent a system of three cross-coupled nonlinear equations with three unknowns, which can be solved using a multi-variable nonlinear solver. We will therefore implement an iterative multi-variable nonlinear solver, called the Multi-Variable Secant Method, to solve for the required control surface deflections δ_A, δ_E, and δ_R, given the required aerodynamic moments, L^A_{ref}, M^A_{ref}, and N^A_{ref}.

The Multi-Variable Secant Method

Given a set of n nonlinear equations in n unknowns given by

$$\begin{aligned} f_1(x_1,x_2,...,x_n) &= 0\\ f_2(x_1,x_2,...,x_n) &= 0 && (6.58)\\ &\vdots\\ f_n(x_1,x_2,...,x_n) &= 0 \end{aligned}$$

We can express the system as a single vector expression

$$\mathbf{f}(\mathbf{x}) = \mathbf{0} \tag{6.59}$$

The system of nonlinear equations can then be solved iteratively using

$$\mathbf{x}(k+1) = \mathbf{x}(k) - \mathbf{J}^{-1}\mathbf{f}(\mathbf{x}(k)) \tag{6.60}$$

where $\mathbf{J}$ is the Jacobian matrix of the system, defined as

$$\mathbf{J} = \begin{bmatrix} \partial f_1/\partial x_1 & \partial f_1/\partial x_2 & \dots & \partial f_1/\partial x_n \\ \partial f_2/\partial x_1 & \partial f_2/\partial x_2 & \dots & \partial f_2/\partial x_n \\ \vdots & \vdots & \ddots & \vdots \\ \partial f_n/\partial x_1 & \partial f_n/\partial x_2 & \dots & \partial f_n/\partial x_n \end{bmatrix} \tag{6.61}$$

and with $\mathbf{x}(0) = \mathbf{x}_0$ as the initial guess of the solution.

Newton's method requires that the Jacobian matrix be explicitly known. The Secant method for a system of nonlinear equations approximates the partial derivatives of the Jacobian matrix with

$$\frac{\partial f_i}{\partial x_j} \approx \frac{f_i(x_1, x_2, ..., x_j + \Delta x, ..., x_n) - f_i(x_1, x_2, ..., x_j, ..., x_n)}{\Delta x} \tag{6.62}$$

Iterative Solution of the Required Control Surface Deflections

Given the target aerodynamic moments L^A_{ref}, M^A_{ref}, and N^A_{ref}, we wish to determine the required control surface deflections δ_A, δ_E, and δ_R. The required control surface deflections are determined by simultaneously solving the nonlinear equations of the wide-envelope aerodynamic model using the multi-variable secant method. For the purposes of applying the multi-variable secant method, it will be useful to represent the wide-envelope aerodynamic model in vector format as follows

$$\mathbf{M^A} = \mathbf{f_M}^A(\rho, \overline{V}, \alpha, \beta, \boldsymbol{\omega}, \boldsymbol{\delta}) \tag{6.63}$$

with

$$\mathbf{M^A} = \begin{bmatrix} L^A & M^A & N^A \end{bmatrix}^T \tag{6.64}$$

$$\boldsymbol{\omega} = \begin{bmatrix} P & Q & R \end{bmatrix}^T \tag{6.65}$$

$$\boldsymbol{\delta} = \begin{bmatrix} \delta_A & \delta_E & \delta_R \end{bmatrix}^T \tag{6.66}$$

$$\mathbf{f_M}^A(\rho, \overline{V}, \alpha, \beta, \boldsymbol{\omega}, \boldsymbol{\delta}) = \begin{bmatrix} \frac{1}{2}\rho\overline{V}^2 SbC_l(\alpha, \beta, P, Q, R, \delta_A, \delta_E, \delta_R) \\ \frac{1}{2}\rho\overline{V}^2 S\bar{c}C_m(\alpha, \beta, P, Q, R, \delta_A, \delta_E, \delta_R) \\ \frac{1}{2}\rho\overline{V}^2 SbC_n(\alpha, \beta, P, Q, R, \delta_A, \delta_E, \delta_R) \end{bmatrix} \tag{6.67}$$

where $\mathbf{M^A}$ is the aerodynamic moment vector, $\boldsymbol{\omega}$ is the body angular rate vector, $\boldsymbol{\delta}$ is the control surface deflection vector, and $\mathbf{f_M}^A$ is the set of nonlinear equations representing the wide-envelope aerodynamic model.

The multi-variable secant method is iteratively executed as shown below. Additional steps have been added to the algorithm to accommodate potential actuator saturations. The approach

is taken to saturate the control surface deflections in such a way that none of the control surfaces "oversaturate", but that one or more actuators just marginally saturate. The marginal saturation is performed so that the direction of the intended control surface deflection vector is maintained, and only its magnitude is reduced.

This approach was motivated by the fact that the control surface deflections produce cross-coupling aerodynamic moments, but that the simultaneous solution of the aerodynamic equations take the cross-coupling moments into account. For example, the ailerons are intended to produce rolling moments but also produce unintended yawing moments, while the rudder is intended to produce yawing moments, but also produces unintended rolling moments. The simultaneous solution takes these cross-coupling moments into account when calculating the required control surface deflection vector to produce a desired aerodynamic moment vector. If the control surfaces are individually saturated without maintaining the direction of the control surface deflection vector, the cross-coupling aerodynamic moments would no longer be cancelled, and unintended aerodynamic disturbance moments would appear.

Step 1: Initialise the estimate of the control surface deflection vector with an initial guess $\boldsymbol{\delta}(0) \leftarrow \boldsymbol{\delta}_0$ at iteration $k = 0$.

$$\begin{aligned} \hat{\delta}_A(0) &\leftarrow \hat{\delta}_{A_0} \\ \hat{\delta}_E(0) &\leftarrow \hat{\delta}_{E_0} \\ \hat{\delta}_R(0) &\leftarrow \hat{\delta}_{R_0} \end{aligned}$$

Step 2: Calculate the aerodynamic moments that would be produced by the current estimate of the control surface deflections.

$$\begin{aligned} \hat{L}^A &\leftarrow \frac{1}{2}\rho\overline{V}^2 SbC_l(\alpha, \beta, P, Q, R, \hat{\delta}_A, \hat{\delta}_E, \hat{\delta}_R) \\ \hat{M}^A &\leftarrow \frac{1}{2}\rho\overline{V}^2 S\bar{c}C_m(\alpha, \beta, P, Q, R, \hat{\delta}_A, \hat{\delta}_E, \hat{\delta}_R) \\ \hat{N}^A &\leftarrow \frac{1}{2}\rho\overline{V}^2 SbC_n(\alpha, \beta, P, Q, R, \hat{\delta}_A, \hat{\delta}_E, \hat{\delta}_R) \end{aligned}$$

Step 3: If the aerodynamic moments are close enough to the desired aerodynamic moments, then return the current estimate of the control surface deflections as the solution.

if $\|L^A_{\text{ref}} - \hat{L}^A\| < \epsilon$ **and** $\|M^A_{\text{ref}} - \hat{M}^A\| < \epsilon$ **and** $\|N^A_{\text{ref}} - \hat{N}^A\| < \epsilon$ **then**
  return $\hat{\delta}_A, \hat{\delta}_E, \hat{\delta}_R$
end if

Step 4: If the aerodynamic moments are not close enough to the desired aerodynamice moments, then determine the approximate Jacobian matrix for the current estimate of the control surface deflections. Determine the partial derivatives with respect to aileron by varying only the aileron deflection

$$\begin{bmatrix} \frac{\partial L}{\partial \delta_A} \\ \frac{\partial M}{\partial \delta_A} \\ \frac{\partial N}{\partial \delta_A} \end{bmatrix} \leftarrow \frac{{\mathbf{f_M}}^A(\ldots, \hat{\delta}_A + \Delta\delta_A, \hat{\delta}_E, \hat{\delta}_R) - {\mathbf{f_M}}^A(\ldots, \hat{\delta}_A, \hat{\delta}_E, \hat{\delta_R})}{\Delta\delta_A}$$

Determine the partial derivatives with respect to elevator by varying only the elevator deflection

$$\begin{bmatrix} \frac{\partial L}{\partial \delta_E} \\ \frac{\partial L}{\partial \delta_E} \\ \frac{\partial N}{\partial \delta_E} \end{bmatrix} \leftarrow \frac{\mathbf{f_M}^A(\ldots, \hat{\delta}_A, \hat{\delta}_E + \Delta\delta_E, \hat{\delta}_R) - \mathbf{f_M}^A(\ldots, \hat{\delta}_A, \hat{\delta}_E, \hat{\delta}_R)}{\Delta\delta_E}$$

Determine the partial derivatives with respect to rudder by varying only the rudder deflection

$$\begin{bmatrix} \frac{\partial L}{\partial \delta_R} \\ \frac{\partial L}{\partial \delta_R} \\ \frac{\partial N}{\partial \delta_R} \end{bmatrix} \leftarrow \frac{\mathbf{f_M}^A(\ldots, \hat{\delta}_A, \hat{\delta}_E, \hat{\delta}_R + \Delta\delta_R) - \mathbf{f_M}^A(\ldots, \hat{\delta}_A, \hat{\delta}_E, \hat{\delta}_R)}{\Delta\delta_R}$$

Concatenate the three column vectors to obtain the approximate Jacobian matrix

$$\mathbf{J} \leftarrow \begin{bmatrix} \frac{\partial L}{\partial \delta_A} & \frac{\partial L}{\partial \delta_E} & \frac{\partial L}{\partial \delta_R} \\ \frac{\partial M}{\partial \delta_A} & \frac{\partial M}{\partial \delta_E} & \frac{\partial M}{\partial \delta_R} \\ \frac{\partial N}{\partial \delta_A} & \frac{\partial N}{\partial \delta_E} & \frac{\partial N}{\partial \delta_R} \end{bmatrix}$$

Step 5: Calculate the next iteration of the control surface deflection vector.

$$\hat{\boldsymbol{\delta}}(k+1) \leftarrow \hat{\boldsymbol{\delta}}(k) + \mathbf{J}^{-1}(\mathbf{M}^{\mathbf{A}}_{\text{ref}} - \hat{\mathbf{M}}^{\mathbf{A}})$$

If the Jacobian matrix is singular, then the algorithm exits and reports that a solution could not be found.

Step 6: Check the new estimate of the control surface vector $\hat{\boldsymbol{\delta}}(k+1)$ to see if any of the control surfaces would saturate. If any of them would, then scale the magnitude of the step vector so that the control surface that would take the largest step relative to its remaining margin would just saturate. First, calculate the intended vector step that would be taken this iteration

$$\boldsymbol{\Delta\delta} = \hat{\boldsymbol{\delta}}(k+1) - \hat{\boldsymbol{\delta}}(k) \tag{6.68}$$

Then, calculate the remaining margins of each control surface to its saturation limit *before* the step is taken

if $\Delta\delta_A(k) > 0$ **then**
  $\Delta\delta_{A_{\text{SAT}}} = \delta_{A_{\max}} - \delta_A(k)$
else
  $\Delta\delta_{A_{\text{SAT}}} = \delta_{A_{\min}} - \delta_A(k)$
end if

if $\Delta\delta_E(k) > 0$ **then**
  $\Delta\delta_{E_{\text{SAT}}} = \delta_{E_{\max}} - \delta_E(k)$
else
  $\Delta\delta_{E_{\text{SAT}}} = \delta_{E_{\min}} - \delta_E(k)$
end if

if $\Delta\delta_A(k) > 0$ **then**
  $\Delta\delta_{R_{\text{SAT}}} = \delta_{R_{\max}} - \delta_R(k)$

else
$\Delta\delta_{R_{\text{SAT}}} = \delta_{R_{\min}} - \delta_R(k)$
end if

Find the control surface that has the smallest remaining margin relative to its intended step component

$$\frac{\Delta\delta_{\text{SAT}}}{\Delta\hat{\delta}} = \min\left(\frac{\Delta\delta_{A_{\text{SAT}}}}{\Delta\hat{\delta}_A}, \frac{\Delta\delta_{E_{\text{SAT}}}}{\Delta\hat{\delta}_E}, \frac{\Delta\delta_{R_{\text{SAT}}}}{\Delta\hat{\delta}_R}\right)$$

Scale the magnitude of the step vector by the smallest ratio of remaining margin to intended step size

$$\boldsymbol{\Delta\hat{\delta}} \leftarrow \frac{\delta_{\text{SAT}}}{\Delta\hat{\delta}}\boldsymbol{\Delta\hat{\delta}} \tag{6.69}$$

Finally, recalculate the control surface deflection vector using the scaled step vector to obtain the appropriately saturated control surface deflections.

$$\hat{\boldsymbol{\delta}}(k+1) \quad \leftarrow \quad \hat{\boldsymbol{\delta}}(k) + \boldsymbol{\Delta\hat{\delta}}$$

If any of the control surfaces have saturated, then the saturated estimate of the control surface deflections are returned as the solution.

Step 7: If the maximum number of iterations have been reached, then the algorithm exits and reports that a solution could not be found. If the maximum number of iterations has not been reached, then repeat from **Step 2**.

Actuator Saturation vs Constrained Optimisation

To accommodate the actuator limits of the control surfaces, the approach was taken to saturate the control surface deflections in such a way that none of the control surfaces "oversaturate", but that one or more actuators just marginally saturate so that the direction of the control surface vector is maintained, and only its magnitude is reduced. This approach was taken in an attempt to reduce the appearance of unintended cross-coupling disturbance moments at the cost of a reduced speed of response.

Instead of saturating the control surface deflections, a more sophisticated approach would be to perform a constrained optimisation of the control surface deflections to minimise the difference between the commanded and achieved aerodynamic moments taking into account the actuator limits. The cost function for the optimisation could be formulated in different ways to represent the aerodynamic moment error to be minimised, for example the cost function could be formulated to minimise the error in the magnitude of the aerodynamic moment vector, or it could be formulated to minimise the error in the direction of the aerodynamic moment vector.

In fact, a better approach may be to minimise the difference between the commanded and achieved angular accelerations, or ultimately to minimise the difference between the desired Lyapunov energy rate and the achieved Lyapunov energy rate. These ideas have not been explored in this thesis, but are recommended as potential avenues for future research.

6.5.3 Sensing the Angular Rates of the Wind Axis System

The Lyapunov-based controller needs the angular rates of the wind axis system relative to the inertial axis system. Fortunately, these angular rates can be determined from the attitude of the body axis system relative to the inertial axis system, and the specific acceleration measurements in the body axis system, both of which are available from onboard inertial sensors.

First, we recognise that the wind axis system represents the orientation and rotational motion of the aircraft's velocity vector in inertial space. The aircraft's velocity vector represents the motion if its centre of mass, and is therefore affected by the external forces (not the moments) acting on the aircraft. The specific forces acting on the aircraft, namely the aerodynamic forces and the thrust forces, are sensed in the body axis using the onboard accelerometers. The gravity force cannot be sensed directly, but since the orientation of the body axis relative to the inertial axis is available from the inertial measurement system, it can be well modelled. Full knowledge of the forces acting on the aircraft are therefore available to us, which means that we are able to "sense" the motion of the wind axis system relative to the inertial axis system.

The following equations express the roll, pitch, and yaw rate of the body axis system relative to inertial space as the sum of the angular rate of the body axis system relative to the wind axes, and the angular rate of the wind axis system relative to inertial space.

$$\begin{bmatrix} P \\ Q \\ R \end{bmatrix} = \begin{bmatrix} 0 & \sin\alpha \\ 1 & 0 \\ 0 & -\cos\alpha \end{bmatrix} \begin{bmatrix} \dot{\alpha} \\ \dot{\beta} \end{bmatrix} + \begin{bmatrix} \cos\alpha\cos\beta & -\cos\alpha\sin\beta & -\sin\alpha \\ \sin\beta & \cos\beta & 0 \\ \sin\alpha\cos\beta & -\sin\alpha\sin\beta & \cos\alpha \end{bmatrix} \begin{bmatrix} P_W \\ Q_W \\ R_W \end{bmatrix} \tag{6.70}$$

where P, Q, and R are the angular rates of the body axis system relative to the inertial axis system (coordinated in the body axis system), $\dot{\alpha}$ and $\dot{\beta}$ are the angle of attack rate and the sideslip angle rate, and P_W, Q_W, and R_W are the angular rates of the wind axis system relative to the inertial axis system (coordinated in the wind axis system).

At steady state, we want the relative angular rate between the body axis system and the wind axis system to be zero. We therefore substitute $\dot{\alpha} = 0$ and $\dot{\beta} = 0$ into equation (6.70) to obtain

$$\begin{bmatrix} P_B^{WI} \\ Q_B^{WI} \\ R_B^{WI} \end{bmatrix} = \begin{bmatrix} \cos\alpha\cos\beta & -\cos\alpha\sin\beta & -\sin\alpha \\ \sin\beta & \cos\beta & 0 \\ \sin\alpha\cos\beta & -\sin\alpha\sin\beta & \cos\alpha \end{bmatrix} \begin{bmatrix} P_W \\ Q_W \\ R_W \end{bmatrix} \tag{6.71}$$

where we have added the superscipts WI and the subscript B to indicate that the resulting angular rates P_B^{WI}, Q_B^{WI}, R_B^{WI} now represent the angular rate of the wind axis system relative to the inertial axis system, but coordinated in the body axis system. These angular rates are in fact the roll rate, pitch rate, and yaw rate references that we wish to provide to the three Lyapunov control laws in equations (6.32), (6.33), and (6.34).

We now recognise that the pitch rate Q_W and the yaw rate R_W of the wind axis system relative to the inertial axis system are the result of the normal and lateral accelerations of the point mass of the aircraft, expressed as

$$Q_W = -\frac{1}{\overline{V}} a_{Wz} \tag{6.72}$$

$$R_W = \frac{1}{\overline{V}} a_{Wy} \tag{6.73}$$

where $\overline{V}$ is the aircraft velocity, and a_{Wz} and a_{Wy} are the total normal and lateral accelerations of the aircraft coordinated in wind axes. We also recognise that the wind axis roll rate P_W is

not so much the result of the accelerations, but is rather constrained to ensure that the wind axis normal unit vector always lies in the plane of symmetry of the body axis system. We will therefore treat it as an additional reference command $P_{W_{\text{ref}}}$, along with the angle of attack and sideslip angle reference commands α_{ref} and β_{ref}.

$$P_W = P_{W_{\text{ref}}} \tag{6.74}$$

We can now substitute equations (6.74), (6.72) and (6.73) into equation (6.71) to obtain an expression for the roll rate, pitch rate, and yaw rate references in terms of the wind-axis roll rate reference, the wind-axis normal acceleration and the wind-axis lateral acceleration.

$$\begin{bmatrix} P_B^{WI} \\ Q_B^{WI} \\ R_B^{WI} \end{bmatrix} = \begin{bmatrix} \cos\alpha\cos\beta & -\cos\alpha\sin\beta & -\sin\alpha \\ \sin\beta & \cos\beta & 0 \\ \sin\alpha\cos\beta & -\sin\alpha\sin\beta & \cos\alpha \end{bmatrix} \begin{bmatrix} P_{W\,\text{ref}} \\ -\frac{1}{\overline{V}} a_{Wz} \\ \frac{1}{\overline{V}} a_{Wy} \end{bmatrix} \tag{6.75}$$

The wind axis normal and lateral accelerations can be obtained from the inertial sensors. The total acceleration of the aircraft in body axes is expressed as

$$\mathbf{a}_B = \mathbf{a}_B^{\Sigma} + \mathbf{a}_B^{G} \tag{6.76}$$

where $\mathbf{a}_B$ is the total acceleration, $\mathbf{a}_B^{\Sigma}$ is the specific acceleration and $\mathbf{a}_B^{G}$ is the gravitational acceleration, all coordinated in the body axes. The specific acceleration is measured directly by the accelerometers of the inertial measurement unit, while the gravitational acceleration can be determined by transforming the gravitational acceleration from the inertial axis system to the body axis using the attitude of the aircraft, which is also provided by the inertial measurement system. The total acceleration in body axes can then be transformed to wind axis using the angle of attack and sideslip angle measured by the anemometric sensors

$$\mathbf{a}_W = (\mathbf{DCM}^{BW})^{-1}\mathbf{a}_B \tag{6.77}$$

where $\mathbf{a}_W$ is the total acceleration coordinated in the wind axes, and $\mathbf{DCM}^{BW}$ is the direction cosine matrix that transforms vectors from wind axes to body axes.

6.5.4 Adding Limited Integral Control

To improve the steady-state tracking performance of the Lyapunov-based controller, and to allow the controller to compensate for offset errors in the aerodynamic model and in the actuator trim settings, integral terms are added to the Lyapunov control law, as follows

$$k_P\dot{P} = -\frac{1}{2}K[k_P(P - P_B^{WI}) \underbrace{+k_{P_I}P_I}] \tag{6.78}$$

$$k_\alpha\dot{\alpha} = -\frac{1}{2}K[k_\alpha(\alpha - \alpha_{\text{ref}}) + k_Q(Q - Q_B^{WI}) \underbrace{+k_{\alpha_I}\alpha_I}] - k_Q\dot{Q} \tag{6.79}$$

$$k_\beta\dot{\beta} = -\frac{1}{2}K[k_\beta(\beta - \beta_{\text{ref}}) + k_R(R - R_B^{WI}) \underbrace{+k_{\beta_I}\beta_I}] - k_R\dot{R} \tag{6.80}$$

where α_I, β_I and P_I are the time integrals of the angle of attack error, the sideslip angle error, and the body axis roll rate error, respectively.

$$\dot{\alpha}_I = \alpha - \alpha_{\text{ref}} \tag{6.81}$$

$$\dot{\beta}_I = \beta - \beta_{\text{ref}} \tag{6.82}$$

$$\dot{P}_I = P - P_B^{WI} \tag{6.83}$$

However, instead of using normal integrators that integrate freely, they are replaced with limited integrators that stop integrating when the reach lower or upper limits, as follows

$$\dot{\alpha}_I = \begin{cases} 0, & \alpha_I = \alpha_{I\text{lower}} \text{ and } (\alpha - \alpha_{\text{ref}}) < 0, \text{ or} \\ & \alpha_I = \alpha_{I\text{upper}} \text{ and } (\alpha - \alpha_{\text{ref}}) > 0 \\ \alpha - \alpha_{\text{ref}}, & \text{otherwise} \end{cases} \tag{6.84}$$

$$\dot{\beta}_I = \begin{cases} 0, & \beta_I = \beta_{I\text{lower}} \text{ and } (\beta - \beta_{\text{ref}}) < 0, \text{ or} \\ & \beta_I = \beta_{I\text{upper}} \text{ and } (\beta - \beta_{\text{ref}}) > 0 \\ \beta - \beta_{\text{ref}}, & \text{otherwise} \end{cases} \tag{6.85}$$

$$\dot{P}_I = \begin{cases} 0, & P_I = P_{I\text{lower}} \text{ and } (P - P_{\text{ref}}) < 0, \text{ or} \\ & P_I = P_{I\text{upper}} \text{ and } (P - P_{\text{ref}}) > 0 \\ P - P_B^{WI}, & \text{otherwise} \end{cases} \tag{6.86}$$

Stopping the integration when the lower or upper integration limits are reached is a form of pre-emptive integrator anti-windup. Furthermore, the integration limits can be chosen so that they are just wide enough for the integral term contributions to compensate for the largest expected steady-state disturbances, while still being narrow enough so that the integrators do not have a significant effect on the transient response or stability of the controller.

6.5.5 Robustness

This section considers the robustness of the Lyapunov controller to model uncertainty. We will argue that the Lyapunov controller should display significant robustness to model uncertainty due to the nature of Lyapunov-based control and based on the architecture of the controller. The components of the Lyapunov-based controller that may be affected by uncertainties in the aircraft model are the Lyapunov control laws, the solution of the required actuator deflections, and the sensing of the wind-axis angular rates. The potential sources of uncertainty are the wide-envelope dynamic model of the aircraft and the sensor measurements used for feedback and gain scheduling.

The sensor measurements can be ruled out as a source of significant uncertainty, because the air data sensors and inertial sensors available on a typical large transport aircraft are sufficiently accurate and have sufficiently wide measurements ranges for the purpose of the Lyapunov controller. Robustness to sensor failures is not considered, since large transport aircraft are equipped with multiple redundant sensors and employ fault detection and isolation strategies to ensure the continued availability of sensor measurements. It is therefore not within the scope of the Lyapunov controller to be robust to sensor failures. It can therefore be argued that no significant uncertainty is introduced in the sensor measurements of the angle of attack, sideslip angle, or angular rates used for feedback control, or in the sensor measurement of the dynamic pressure used for gain scheduling, or in the sensor measurements of the aircraft attitude and specific accelerations that are used to sense the wind-axis angular rates.

The only significant source of model uncertainty that needs to be considered is therefore the uncertainty in the wide-envelope dynamic model of the aircraft. The potential sources of uncertainty in the aircraft model are the mass, centre of gravity location, and moment of inertia of the body, the wide-envelope aerodynamic model used to calculate the aerodynamic moments, and the moments produced by the engine thrust. The mass, centre of gravity location, and moment of inertia can again be ruled out as sources of significant uncertainty, since they are carefully monitored on large transport aircraft, and are even controlled by actively redistributing the fuel mass inside the aircraft. The moments produced by the engine thrust are also ruled out, since they will be treated as disturbances to be rejected by the feedback control law, rather

than an uncertainty in the model. The only significant source of model uncertainty that remains is therefore the wide-envelope aerodynamic model.

The Lyapunov control law component is robust to uncertainty in the wide-envelope aircraft model, since it does not use a dynamic model containing forces or moments, and only deals with kinematic variables such as angle of attack, sideslip angle, angular rates, and angular accelerations. The kinematic model used by the Lyapunov control law describes the geometry of the motion, and therefore inherently contains very little model uncertainty. The dynamic model that describes the effects of the aerodynamic forces and moments, and typically contains the most model uncertainty, is not used by the Lyapunov control law. It receives angle of attack and sideslip angle references from the pilot or autopilot, the body angular rate references from the component that senses the wind-axis angular rates, the angle of attack and sideslip angle feedback from the air data sensor measurements, and the body angular rate feedback from the inertial sensor measurements. The time derivatives of the angle of attack and sideslip angle are obtained through filtered differentiation of the angle of attack and sideslip angle sensor measurements. The output of the Lyapunov control law is the required angular accelerations, which are also kinematic variables.

The component that senses the wind-axis angular rates is also robust to uncertainty in the wide-envelope aircraft model, since it also does not use a dynamic model containing forces or moments, and also only deals with kinematic variables. The component calculates the wind-axis angular rates using sensor measurements of the angle of attack, sideslip angle, and normal and lateral translational accelerations. Although the component uses the normal and lateral translational accelerations of the aircraft, they are not obtained from a dynamic model, but are obtained directly from the inertial sensors. (The total translational acceleration is obtained by adding the specific acceleration measured by the acceleration sensors and the gravitational acceleration calculated from the accurate attitude provided by the inertial sensors.) The output of the component is the angular rates of the wind axis system, coordinated in the aircraft body axis system.

The only potential robustness risk that the Lyapunov-based controller has, is that the component that calculates the required actuator deflections depends on the dynamic model of the aircraft's rotational motion. The dynamic model uses the wide-envelope aerodynamic model to calculate the aerodynamic moments produced, and then uses the moment of inertia to calculate the resulting angular accelerations. The aerodynamic model uses the dynamic pressure, the aerodynamic sizing constants, and nonlinear aerodynamic coefficient functions that take the angle of attack, the sideslip angle, the body angular rates, and the actuator deflections as input variables. As was argued before, the uncertainty does not lie in the sensor measurements of the anemometric or inertial input variables, or in the moment of inertia of the aircraft, but lies in the nonlinear aerodynamic coefficient functions of the wide-envelope aerodynamic model. Modelling errors in the nonlinear aerodynamic coefficient functions will cause the actual aerodynamic moments to deviate from the desired aerodynamic moments.

Fortunately, the Lyapunov-based nature of the controller makes it inherently robust to uncertainties in the aerodynamic model, because if the actual aerodynamic moments are close enough to the desired aerodynamic moments so that the time derivative of the Lyapunov energy is still negative, then the controller will still provide asymptotic stability. And as the modelling error decreases, the controller will again approach exponential stability. Since the requirements for exponential stability are much more strict than the requirements for asymptotic stability, it is expected that the Lyapunov-based controller will be able to tolerate significant uncertainties in the aerodynamic moments produced.

6.6 Simulation Results

The Lyapunov-based controller was tested on the simulation model of the NASA Generic Transport Model. Simulation tests were performed to verify its ability to perform active angular rate recovery and aerodynamic envelope recovery, and to verify its ability to perform angle of attack, sideslip angle, and wind-axis roll rate command tracking in an extended aerodynamic envelope.

6.6.1 Setup

The simulation tests were performed using the full NASA GTM Simulink model made available by NASA for design and analysis purposes. The Lyapunov-based controller was implemented using a combination of Simulink blocks and embedded Matlab scripts. The Lyapunov controller uses sensor measurements of the angle of attack, sideslip angle, dynamic pressure, and true airspeed from air data sensors, as well as sensor measurements of the three-axis specific accelerations and the gross attitude (bank angle and pitch angle) of the aircraft from the inertial measurement system.

6.6.2 High Angular Rate and Aerodynamic Envelope Recovery

This section presents the simulation results for active angular rate recovery and active aerodynamic envelope recovery. The performance of the Lyapunov-based controller was verified for the following out-of-envelope initial conditions:

- Active Angle of Attack Recovery
- Active Angle of Attack and Sideslip Angle Recovery
- Active Angular Rate Recovery and Aerodynamic Envelope Recovery

For each recovery scenario, a comprehensive set of simulation results is presented to demonstrate the Lyapunov controller's performance over the entire domain of operation, and the results are then discussed in more detail.

Active Angle of Attack Recovery

A comprehensive set of simulation results for active angle of attack recovery are shown in figures 6.6 to 6.10. The aircraft state was initialised with angles of attack from -5 to +60 degrees, and with all other flight variables inside the flight envelope (zero sideslip angle, zero angular rates, zero flight path angle, zero bank angle, and typical airspeed). The angle of attack reference for the Lyapunov controller was set to 4 degrees, and the sideslip angle and wind-axis roll rate references were both set to zero.

The angle of attack and sideslip angle responses are shown in figure 6.6. For all initial angles of attack from -5 to +60 degrees, the angle of attack is successfully recovered to 4 degrees, while the sideslip angle is kept close to zero. The transient response is smooth with good damping and a good speed of response. The transient response exhibits no oscillations and no overshoot, and has a settling time comfortably less than a second. In steady state, both the angle of attack and the sideslip angle track their references with zero steady-state error. A very small transient disturbance may be observed in the sideslip angle response (if one were to zoom in), but it is generally less than 0.1 degrees. (Note that the scales of the plots have intentionally been left "zoomed out" to visually convey the magnitude of the variables relative to their minimum and maximum values.)

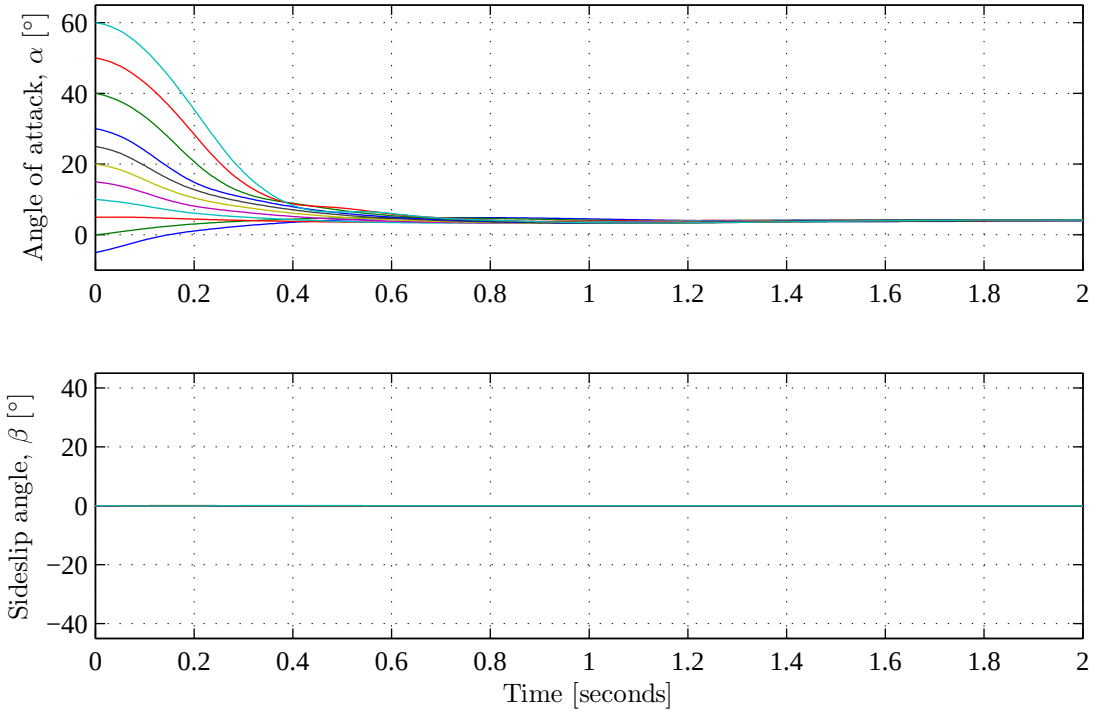


Figure 6.6: Active angle of attack recovery: angle of attack and sideslip angle vs. time

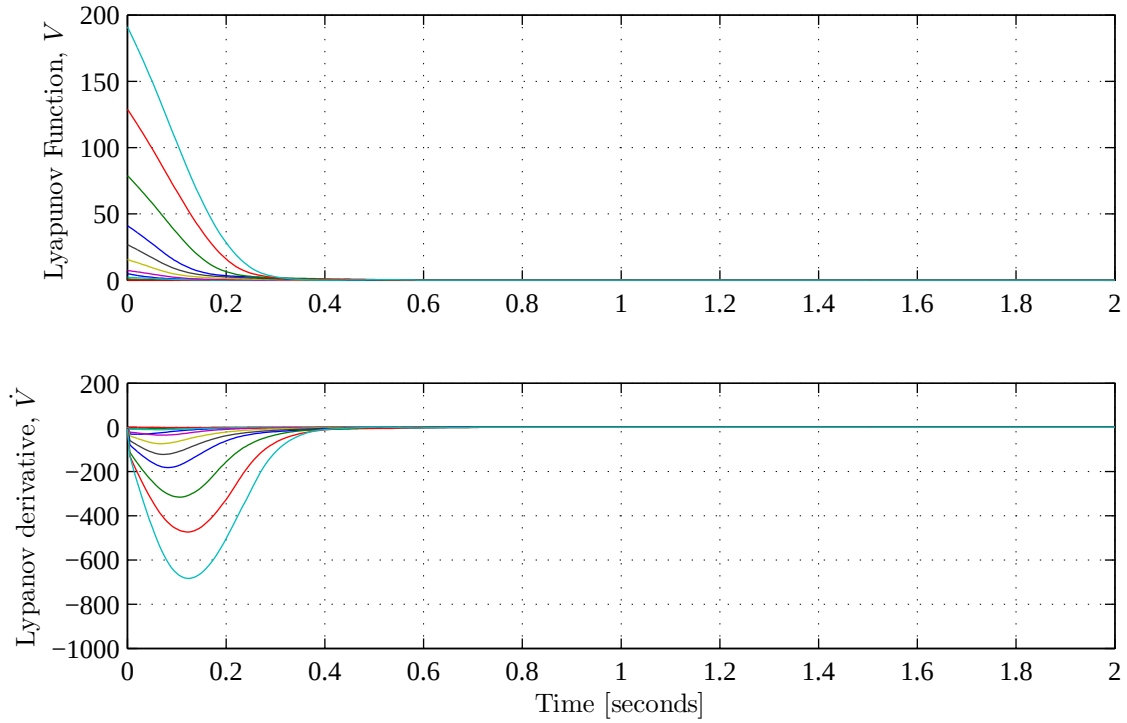


Figure 6.7: Active angle of attack recovery: Lyapunov energy and energy rate vs. time

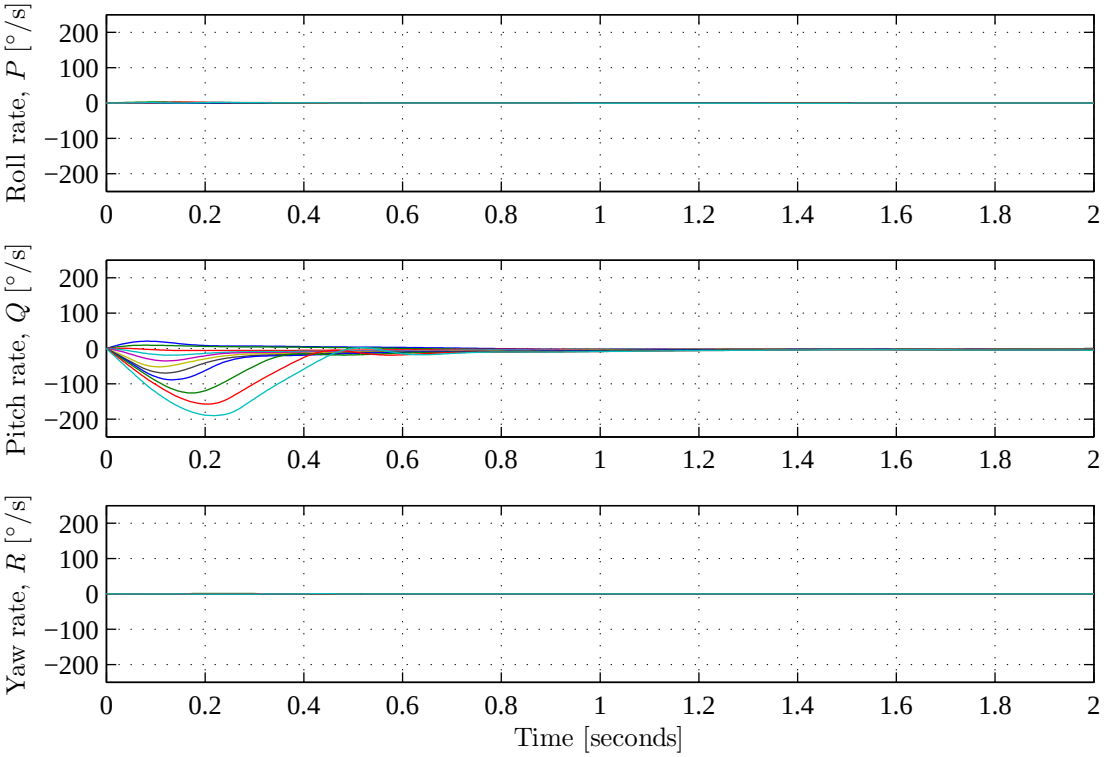


Figure 6.8: Active angle of attack recovery: body angular rates vs. time

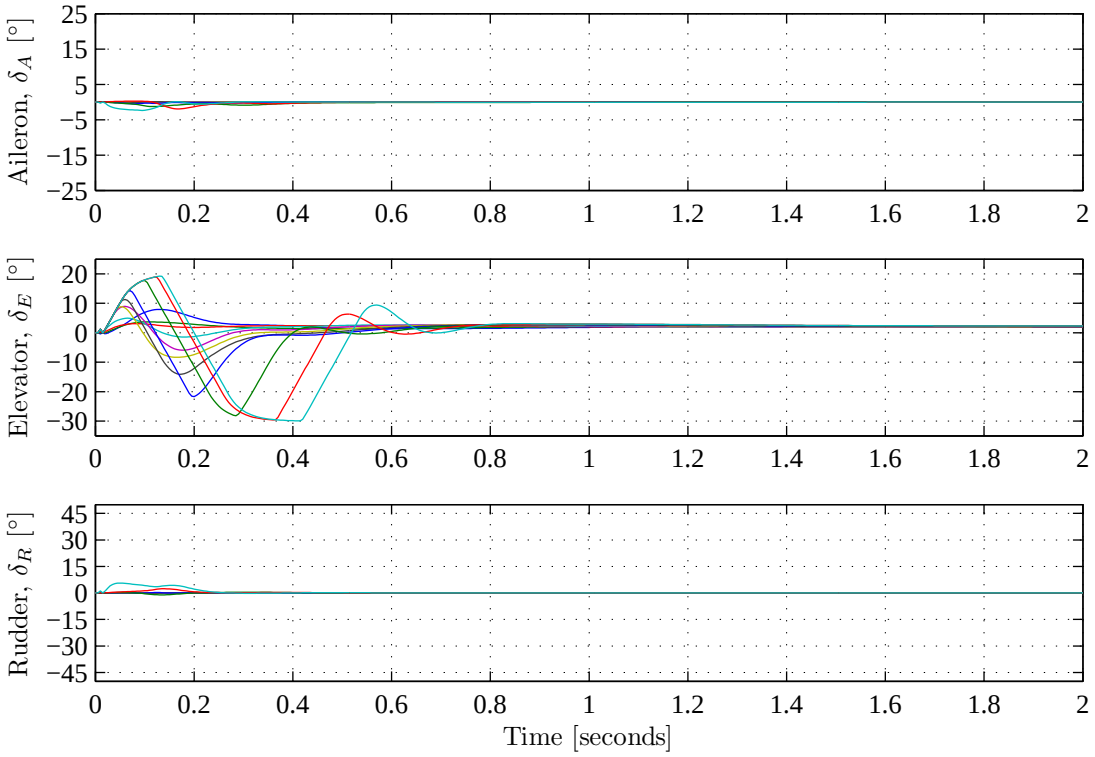


Figure 6.9: Active angle of attack recovery: control surface deflections vs. time

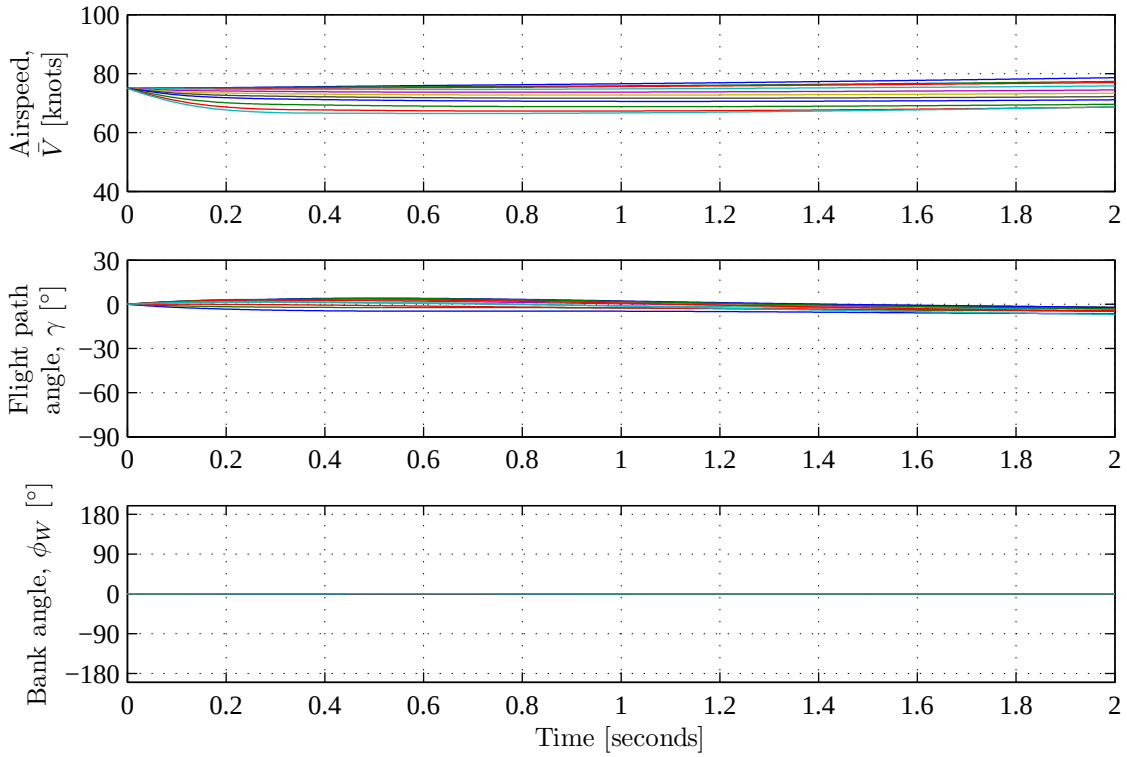


Figure 6.10: Active angle of attack recovery: airspeed, flight path angle, and bank angle vs. time

The body angular rate responses are shown in figure 6.8. A high transient pitch rate is observed, but is expected, since pitch rate is used to recover the angle of attack. Small transient disturbances may be observed in the roll rate and pitch rate responses (if one were to zoom in), but the disturbances are quite small (roll rate < 4 deg/s and yaw rate < 1 deg/s). In steady state, the angular rates remain close to zero.

The control surface responses are shown in figure 6.9. The actuator responses show that the elevators are used to perform the angle of attack recovery, and that there is very little usage of the ailerons and the rudders. The elevators just saturate during the first part of the angle of attack recovery due to the fact that the Lyapunov controller asks for a pitching moment greater than what the elevators can physically provide. Slew rate limiting of the elevators are also clearly observed. In steady state, the elevators settle to the trim deflection required to maintain the angle of attack at the given airspeed. Very small transients are observed in the aileron and rudder deflections (if one were to zoom in), as they work to reject the small transient disturbances in the sideslip angle and in the body angular rates.

The instantaneous Lyapunov energy V and Lyapunov energy rate $\dot{V}$ responses are shown in figure 6.7. The Lyapunov energy responses show that the Lyapunov energy is always positive and that its time response resembles an exponential decay with a time constant of between 0.1 and 0.15 seconds. The Lyapunov control law was designed to provide exponential stability by making the Lyapunov energy rate negative and proportional to Lyapunov energy itself. The proportional constant was chosen as $K = 10$ which means that we expect the energy to decay exponentially with a time constant of 0.1 seconds. The reason that the simulated time constant is longer than the designed time constant, is the fact that the actuators saturated during the first part of the angle of attack recovery for high initial angles of attack. When the actuators saturate, the Lyapunov controller is no longer able to make the Lyapunov energy rate proportional to the Lyapunov energy, but is at least still able to make the Lyapunov energy

rate negative. This means that while the actuators saturate, the Lyapunov controller provides asymptotic stability, and when the actuators come out of saturation, the Lyapunov controller provides exponential stability again.

The time histories of the airspeed, flight path angle and bank angle are shown in figure 6.10, to illustrate that the aircraft centre of mass still experiences translational motion while the rotational motion about the centre of mass is recovered by the Lyapunov controller. In accordance with the proposed integrated flight envelope recovery strategy, the Lyapunov-based inner-loop controller is only responsible for recovering the angle of attack, the sideslip angle, and the body angular rates. The gross attitude, the flight path angle, and the airspeed of the aircraft is left uncontrolled at this stage, as the recovery of the translational motion of the aircraft will be the responsiblity of an outer-loop controller.

From the responses in figure 6.10, it can be seen that the translational motion of the aircraft changes very little over the time scale of the active angle of attack recovery. The initial angle of attack does affect the airspeed and the flight path angle during the first part of the response, reducing the airspeed slightly due to the higher initial induced drag and increasing the flight path angle slightly due to the initial higher lift. However, after the angle of attack is recovered in the first second, the aircraft continues flying in a straight line with both the airspeed and flight path angle remaining essentially constant. The bank angle is undisturbed by the angle of attack recovery and remains close to its initial wings level state.

Active Angle of Attack & Sideslip Angle Recovery

A comprehensive set of simulation results for active combined angle of attack and sideslip angle recovery are shown in figures 6.11 to 6.15. The aircraft state was initialised with angles of attack from -5 to +60 degrees and sideslip angles from -45 to +45 degrees, and with all other flight variables inside the flight envelope (zero angular rates, zero flight path angle, zero bank angle, and typical airspeed). The angle of attack reference for the Lyapunov controller was set to 4 degrees, and the sideslip angle and wind-axis roll rate references were both set to zero.

The angle of attack and sideslip angle responses are shown in figure 6.11. For all initial angles of attack and sideslip angles in the domain, both the angle of attack and the sideslip angle are successfully recovered to 4 degrees and 0 degrees respectively. The transient response is smooth with good damping and a good speed of response. The transient response exhibits no oscillations and no overshoot, and has a settling time comfortably less than a second. In steady state, both the angle of attack and the sideslip angle track their references with zero steady-state error.

The body angular rate responses are shown in figure 6.13. High transient angular rates are observed in all three axes. The high transient pitch and yaw rates are expected, since pitch rate is used to recover angle of attack, and yaw rate is used to recovery sideslip angle. The high transient roll rate is the result of the large initial rolling moment induced by the high initial sideslip angle. In steady state, the body angular rates are recovered, with the body-axis roll rate indirectly tracking the zero wind-axis roll rate reference, and the body-axis pitch rate and body-axis yaw rate indirectly tracking the natural wind-axis pitch rate and wind-axis yaw rate due to the normal and lateral translational accelerations experienced by the aircraft centre of mass.

The control surface response are shown in figure 6.14. The actuator responses show that the ailerons, elevators and rudder are all used to perform the combined angle of attack and sideslip angle recovery. For high sideslip angles, the ailerons saturate during the first part of the aerodynamic recovery to counter the large rolling moment induced by the high initial sideslip angle. The elevators and rudder also saturate during the first part of the response, for high initial angles of attack and high initial sideslip angles respectively. Slew rate limiting for all the actuators are also clearly observed.

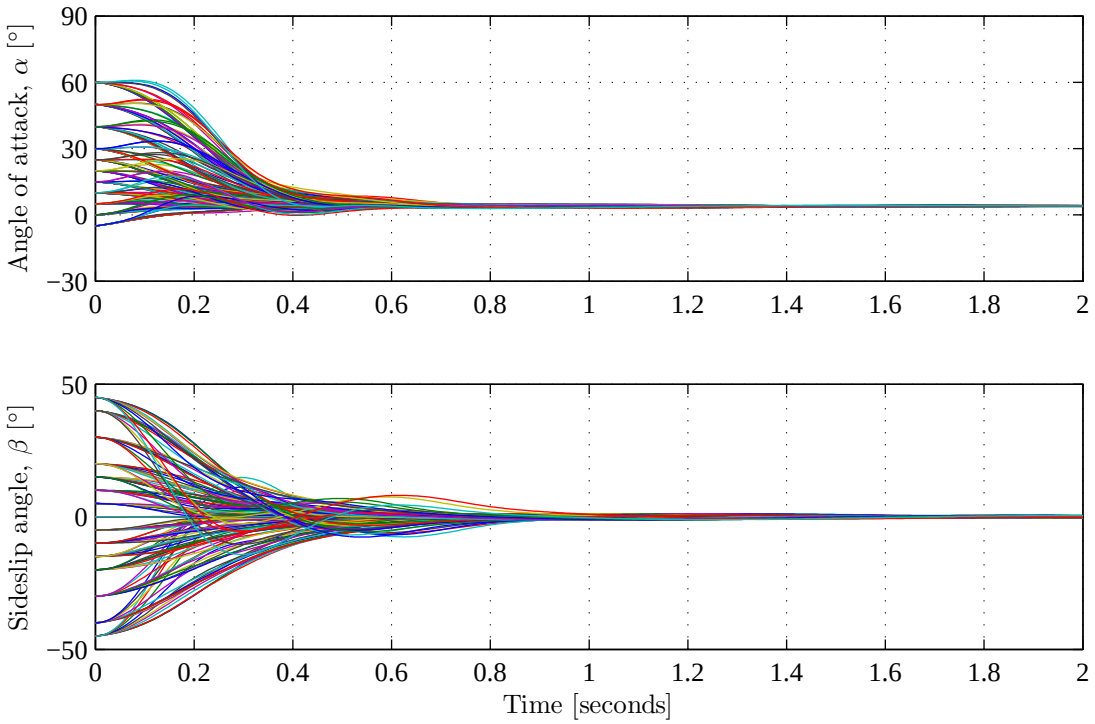


Figure 6.11: Active angle of attack and sideslip angle recovery: angle of attack and sideslip angle vs. time

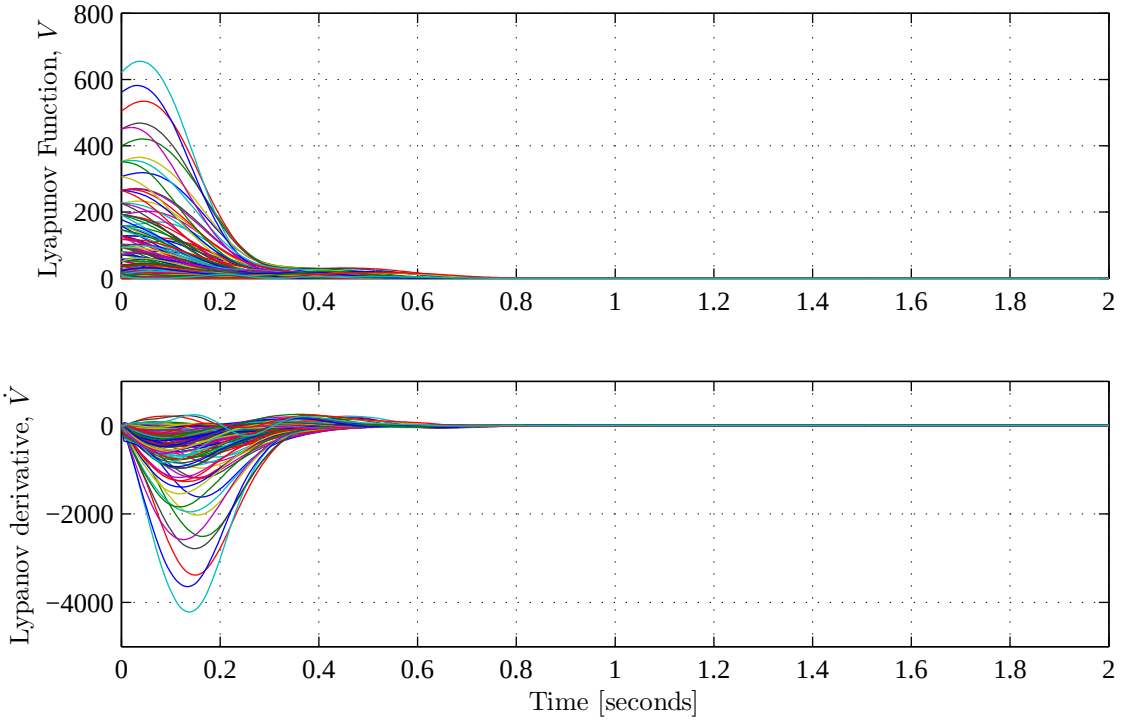


Figure 6.12: Active angle of attack and sideslip angle recovery: Lyapunov energy and energy rate vs. time

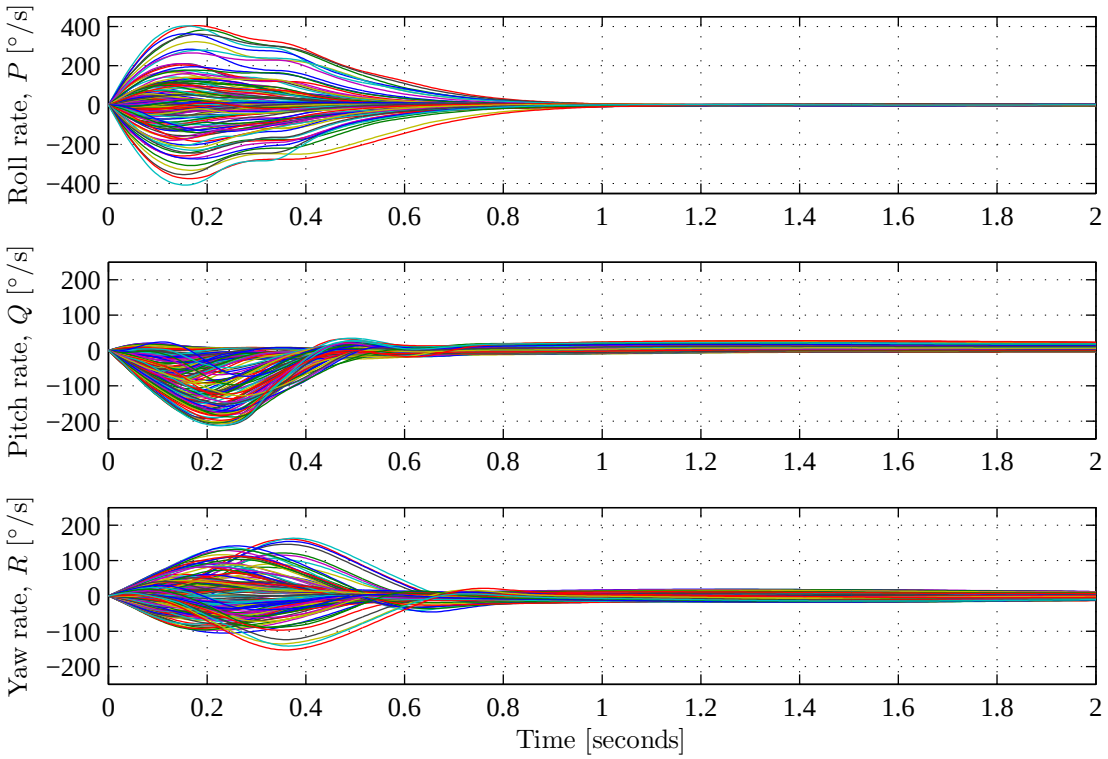


Figure 6.13: Active angle of attack and sideslip angle recovery: body angular rates vs. time

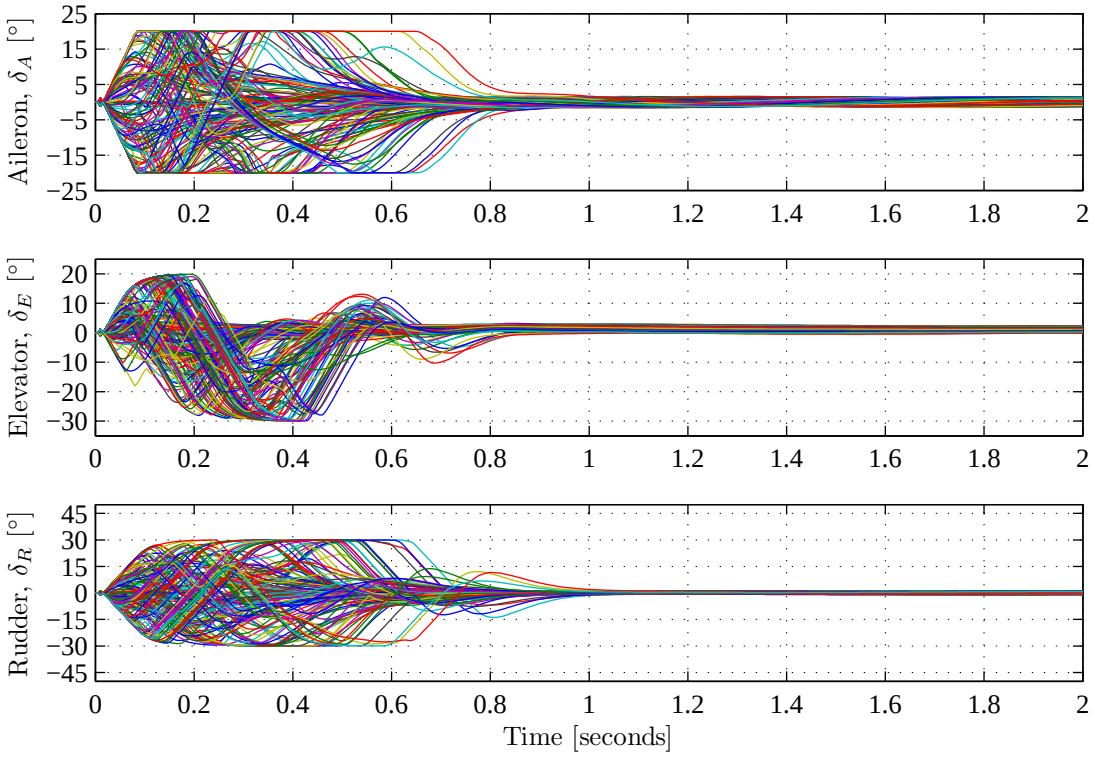


Figure 6.14: Active angle of attack and sideslip angle recovery: control surface deflections vs. time

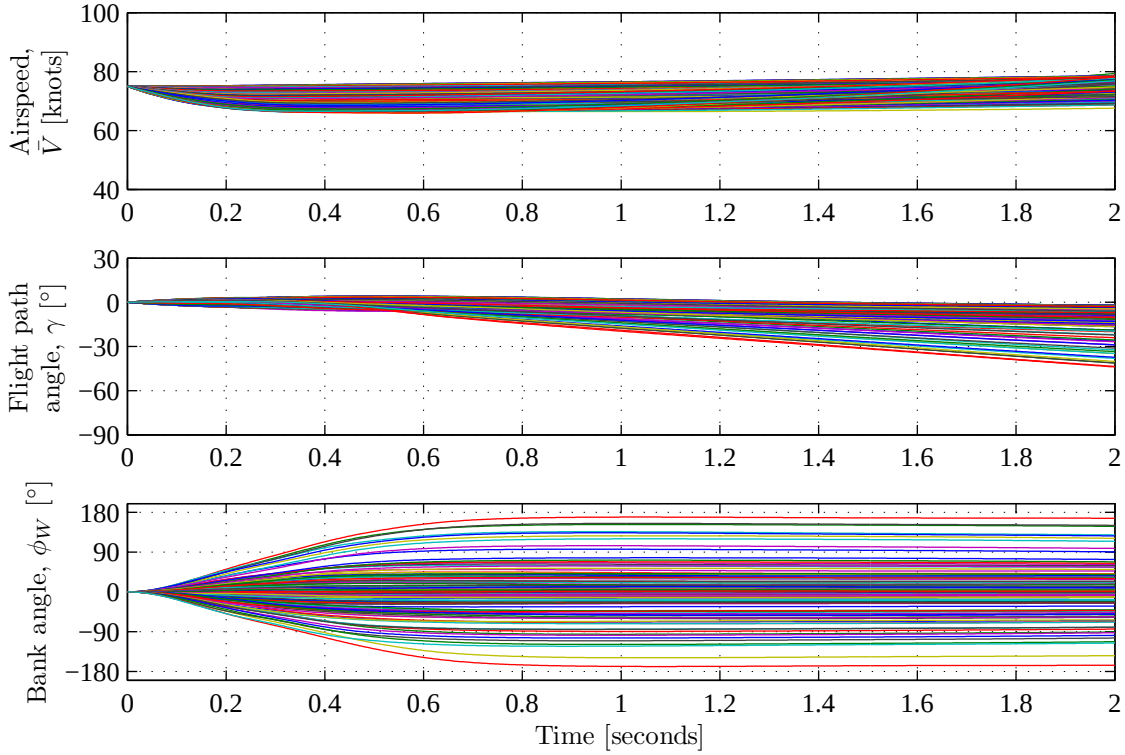


Figure 6.15: Active angle of attack and sideslip angle recovery: airspeed, flight path angle, and bank angle vs. time

In steady state, the control surfaces all settle to the trim deflections required to maintain the angle of attack and sideslip angle at the given airspeed.

The instantaneous Lyapunov energy V and Lyapunov energy rate $\dot{V}$ responses are shown in figure 6.12. The Lyapunov energy responses show that the Lyapunov energy is always positive and that its time response again resembles an exponential decay with a time constant of between 0.1 and 0.2 seconds. For most of the trajectories, the Lyapunov energy rate is always negative, however for some high angle of attack and high sideslip angle initial conditions, the Lyapunov energy rate may become positive. This occurs when the actuators saturate and the aerodynamic moments are dominated by the static moments due to angle of attack and sideslip angle, and the dynamic moments due to high angular rates. When the actuators saturate, the Lyapunov controller is no longer able to provide exponential stability by making the Lyapunov energy rate proportional to the Lyapunov energy, and then attempts to provide asymptotic stability by making the Lyapunov energy rate at least negative. However, in some cases where the control moments are overpowered by the static moments and dynamic moments, the Lyapunov energy rate may become positive. It is important to note that the system does not necessarily become unstable when the Lyapunov energy rate becomes positive, it only means that the stability of the system is no longer guaranteed. From the simulation results it can be seen that all the trajectories converge to the desired angle of attack and sideslip angle, including the trajectories for which the Lyapunov energy rate becomes positive in the transient.

The time histories of the airspeed, flight path angle and bank angle are shown in figure 6.15, to illustrate that the aircraft centre of mass still experiences translational motion while the rotational motion about the centre of mass is recovered by the Lyapunov controller. Once again, the Lyapunov controller is only responsible for recovering the aerodynamic envelope and the body angular rates, and the gross attitude and translational motion of the aircraft is left uncontrolled. However, it is still interesting to observe what happens with the aircraft

translational motion during the aerodynamic envelope recovery. During the first part of the response, the initial higher angle of attack again causes the airspeed to reduce slightly due to the higher initial induced drag and the flight path angle to increase slightly due to the higher initial lift. More significantly, the bank angle is disturbed to high bank angles up to 180 degrees for the trajectories that have high initial sideslip angles. It is also interesting to observe that for all trajectories, the bank angle eventually reaches a constant steady-state value after the aerodynamic envelope is recovered, due to the fact that the Lyapunov controller arrests the wind-axis roll rate and regulates it to zero. (We note that all the bank angle trajectories run parallel after about one second.) For the trajectories with unusual bank angles, the flight path angle becomes increasingly negative over time, due to the fact that the lift vector is in the horizontal plane or inverted, and therefore does not sufficiently oppose the gravity force to maintain level flight. The airspeed then also steadily increases under the influence of gravity, due to the nose-down flight path angle of the aircraft, and the reduced drag at the lower angle of attack. However, these uncontrolled deviations in flight path angle and bank angle are not a serious cause for concern yet, since they will eventually be controlled by an outer-loop controller that uses the Lyapunov-based inner-loop controller to recover them.

Active Angular Rate Recovery

A comprehensive set of simulation results for active angular rate and aerodynamic envelope recovery are shown in figures 6.16 to 6.20. Monte Carlo simulations were performed with the aircraft state randomly initialised with angles of attack from -5 to +60 degrees, sideslip angles from -45 to +45 degrees, and with the aircraft spinning about its velocity vector with wind-axis roll rates from -180 to +180 degrees per second. All other flight variables were initialised inside the flight envelope (shallow negative flight path angle, zero bank angle, and typical airspeed). The angle of attack reference for the Lyapunov controller was set to 4 degrees, and the sideslip angle and wind-axis roll rate references were both set to zero. A set of 1000 Monte Carlo simulations were performed with the initial angle of attack, sideslip angle, and wind-axis roll rate uniformly distributed in the given ranges.

The angle of attack and sideslip angle responses, and the wind-axis roll rate response, are shown in figure 6.16. For all initial states, both the angle of attack and the sideslip angle are successfully recovered to 4 degrees and 0 degrees, and the wind-axis roll rate is recovered to zero degrees per second. A high transient wind-axis roll rate is observed for trajectories with high initial sideslip angles, due to the large initial rolling moment induced by the sideslip angle. However, even for extreme initial angles of attack, sideslip angles, and aircraft spin rates, the angular rates and aerodynamic envelope of the aircraft are eventually recovered with acceptable transients and a good speed of response. The transient response remains relatively smooth, exhibiting only a gentle low frequency oscillation, and the speed of response has a worst-case settling time of less than 1.5 seconds. In steady state, both the angle of attack and the sideslip angle track their references with small steady-state errors, and the wind-axis roll rate tracks its reference with zero-steady state error.

The body angular rate responses are shown in figure 6.18. For all initial states, the body angular rates are successfully reduced and recovered. In steady state, the body-axis roll rate indirectly tracks the zero wind-axis roll rate reference, and the body-axis pitch rate and body-axis yaw rate indirectly track the natural wind-axis pitch rate and wind-axis yaw rate due to the normal and lateral translational accelerations experienced by the aircraft centre of mass.

The control surface response are shown in figure 6.19. The actuator responses show that the ailerons, elevators and rudder are all used to perform the angular rate and aerodynamic envelope recovery. For high spin rates and sideslip angles, the control surfaces all saturate while countering the large wind-axis roll rate and the large rolling moment induced by the high initial angle of attack. Slew rate limiting for all the actuators is also clearly observed.

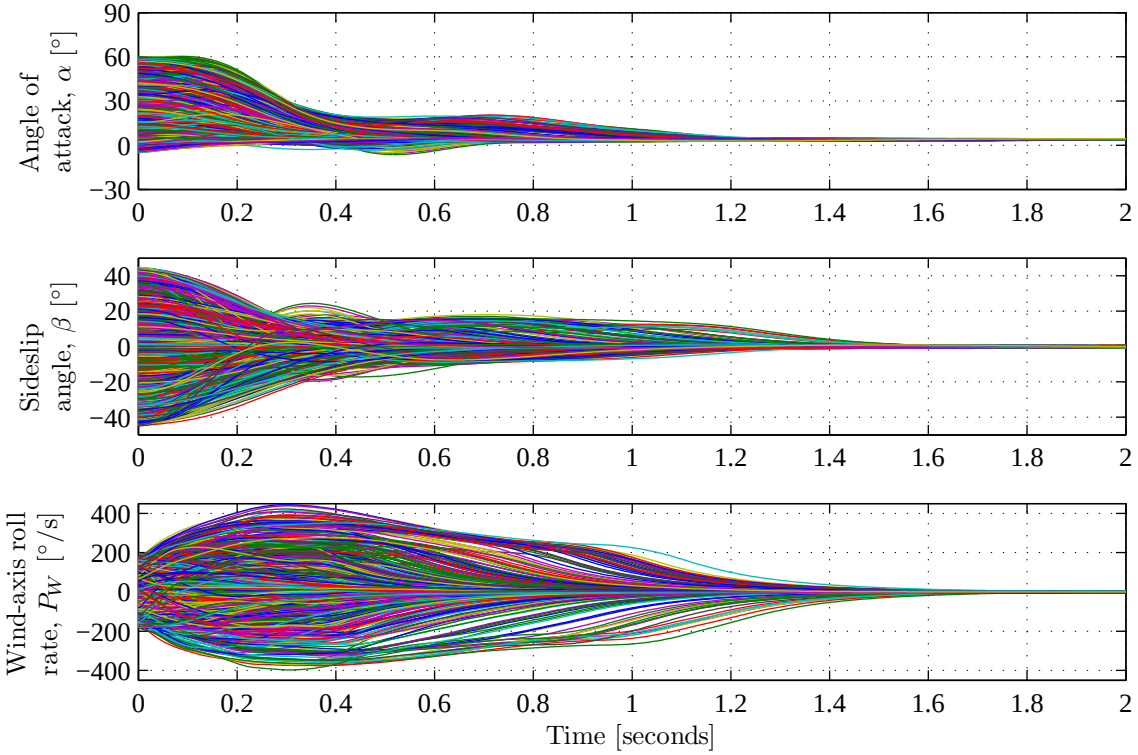


Figure 6.16: Active angular recovery: angle of attack, sideslip angle, and wind-axis roll rate vs. time

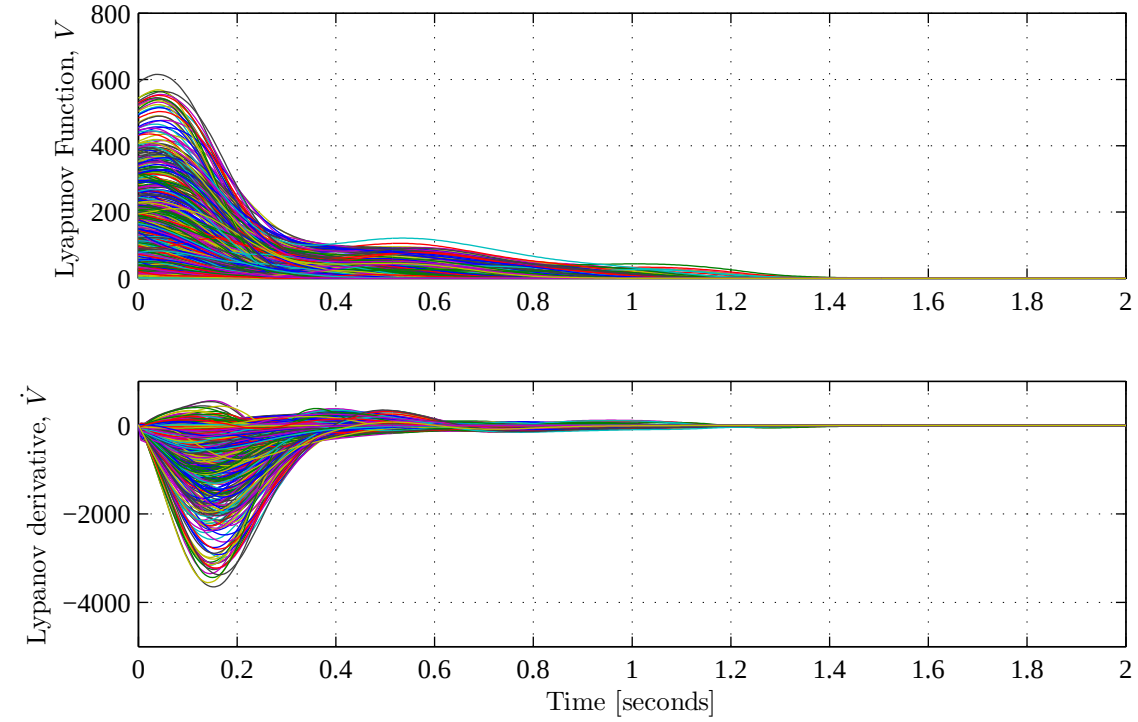


Figure 6.17: Active angular rate recovery: Lyapunov energy and energy rate vs. time

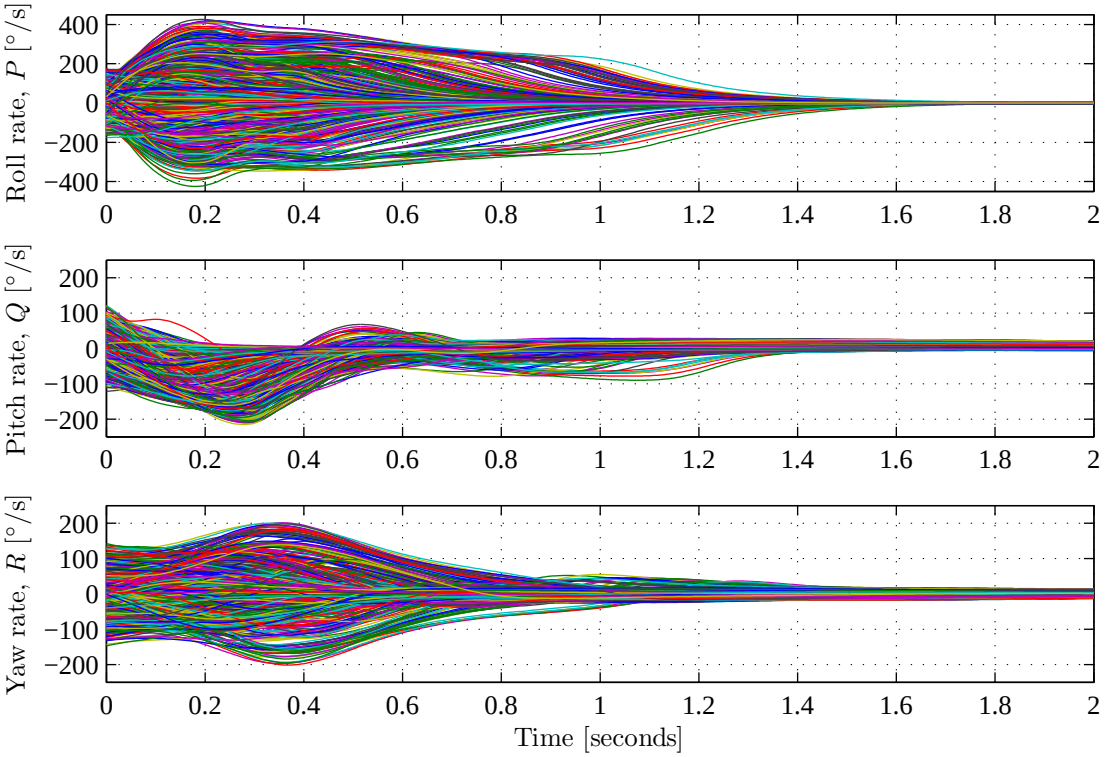


Figure 6.18: Active angular rate recovery: body angular rates vs. time

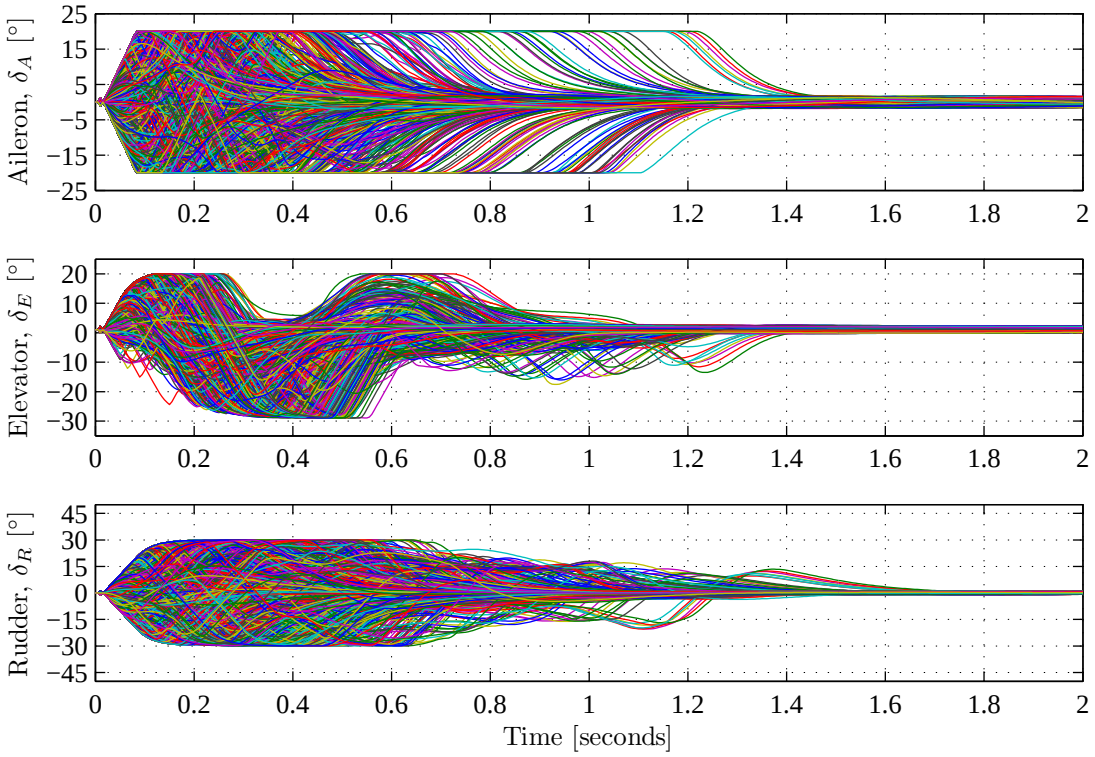


Figure 6.19: Active angular rate recovery: control surface deflections vs. time

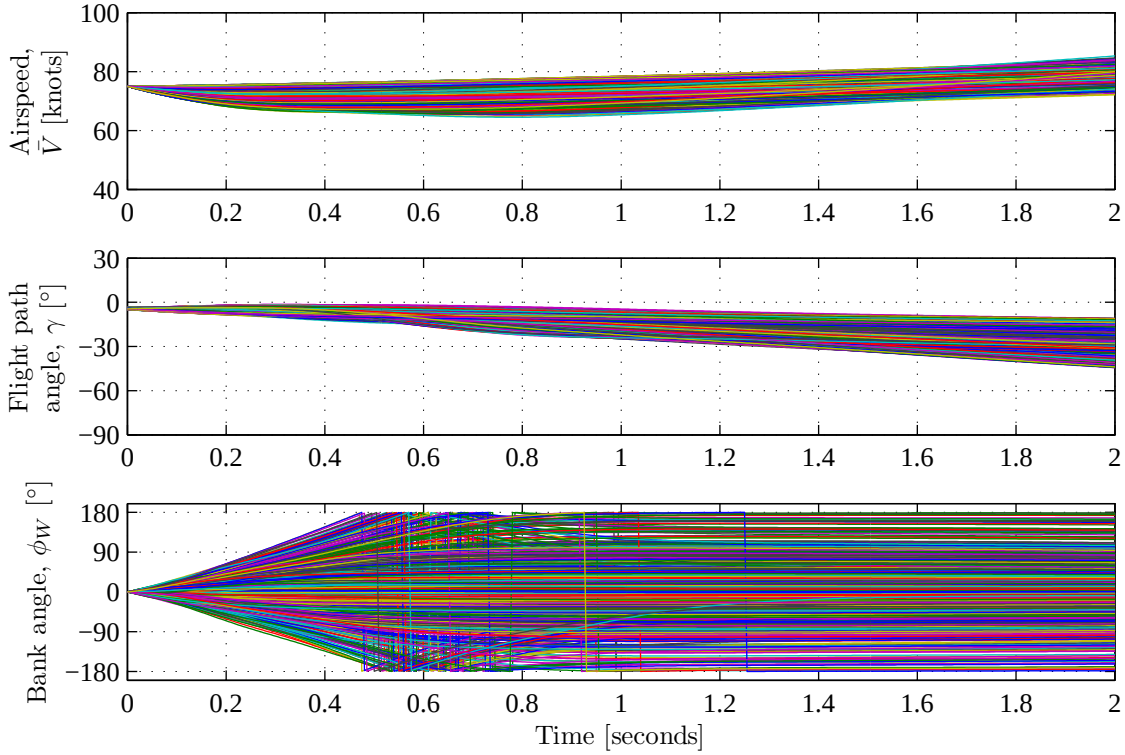


Figure 6.20: Active angular rate recovery: airspeed, flight path angle, and bank angle vs. time

In steady state, the control surfaces all settle to the trim deflections required to maintain the angle of attack and sideslip angle at the given airspeed.

The instantaneous Lyapunov energy V and Lyapunov energy rate $\dot{V}$ responses are shown in figure 6.17. The Lyapunov energy responses show that the Lyapunov energy is always positive and that it eventually decays to zero with a worst-case settling time less than 1.5 seconds. For most of the trajectories, the Lyapunov energy rate is always negative, however for some cases with high angles of attack, high sideslip angles and/or high angular rates, the Lyapunov energy rate may become positive. As before, this occurs when the actuators saturate and the aerodynamic moments are dominated by the static moments due to angle of attack and sideslip angle, and the dynamic moments due to high angular rates. As before, the Lyapunov controller provides exponential stability when the actuators do not saturate, and attempts to provide at least asymptotic stability when the actuators do saturate. As stated before, a positive Lyapunov energy rate does not mean that the system becomes unstable, it only means that the stability is no longer guaranteed. However, the simulation results show that the Lyapunov controller is able to successfully recover the angular rates and aerodynamic envelopes from all the initial states, even those for which the Lyapunov energy rate becomes positive in the transient.

The time histories of the airspeed, flight path angle and bank angle are shown in figure 6.20. For initial states with high wind-axis roll rates and/or high sideslip angles, the bank angle is severely disturbed and even wraps around -180 and +180 degrees one or more times as the aircraft effectively performs "barrel rolls" due to the high initial roll rate. However, for all trajectories the rolling motion is eventually arrested by the Lyapunov controller, and the bank angle eventually reaches a constant steady-state value. We note that the bank angle trajectories all run parallel after about 1.5 seconds. For the trajectories with unusual bank angles, the flight path angle becomes increasingly negative over time, due to the fact that the lift vector is in the horizontal plane or inverted, and therefore does not sufficiently oppose the gravity force to maintain level flight. The airspeed then also steadily increases under the influence of gravity,

due to the nose-down flight path angle of the aircraft, and the reduced drag at the lower angle of attack. Again, these uncontrolled deviations in flight path angle and bank angle are not a serious cause for concern, since they will eventually be controlled by an outer-loop controller that uses the Lyapunov-based inner-loop controller to recover them.

6.6.3 Command Tracking

This section presents the simulation results for active command tracking performance in an extended aerodynamic envelope. The command tracking performance of the Lyapunov-based controller was verified for the following combinations of angle of attack, wind-axis roll rate, and sideslip angle commands:

- Angle of Attack Command Tracking
- Angle of Attack and Wind-Axis Roll Rate Command Tracking
- Angle of Attack, Wind-Axis Roll Rate, and Sideslip Angle Command Tracking

For each command tracking scenario, a comprehensive set of simulation results is presented to demonstrate the Lyapunov controller's performance over the entire domain of operation, and the results are then discussed in more detail.

Angle of Attack Command Tracking

The simulation results for angle of attack command tracking in an extended envelope are shown in figures 6.21 to 6.24. The aircraft was initialised with all flight variables inside the flight envelope (trim angle of attack, zero sideslip angle, zero angular rates, zero bank angle, zero flight path angle, and typical airspeed). The angle of attack reference for the Lyapunov controller was initially set to 4 degrees, and the sideslip angle and wind-axis roll rate references were both set to zero. At t = 0.5 seconds, the angle of attack reference is stepped to values of 0, 8, 12, 16 and 21 degrees, and at t = 2.5 seconds the angle of attack reference is stepped back to 4 degrees.

The angle of attack, wind-axis roll rate, and sideslip angle responses are shown in figure 6.21. The step responses shows good command tracking of the angle of attack. The transient response looks good with overshoot of about 5% and a settling time of about 1.3 seconds, and the steady-state error is less then 0.1 degrees. The Lyapunov controller is able to track a high angle of attack command of 21 degrees, and is also able to return the angle of attack to a normal angle of attack of 4 degrees. The wind-axis roll rate and sideslip angle are kept close to zero during the angle of attack tracking.

The Lyapunov energy and Lyapunov energy rate responses are shown in figure 6.22. The responses show that the Lyapunov energy is always positive and that there are steps in the Lyapunov energy corresponding to each angle of attack reference step. Following each step, the Lyapunov energy decays back to zero with a response that resembles an exponential response. The responses are explained as follows: When an angle of attack step command is given, the equilibrium state of the system is instantaneously changed, which means that the current state of the system is no longer the equilibrium. The Lyapunov energy of the system therefore also instantenously steps to a non-zero positive value to reflect the fact the system is now effectively perturbed from its new equilibrium state. The Lyapunov controller then commands the actuators to reduce the Lyapunov energy so that it decays to the zero energy represented by the new equilibrium state. When the Lyapunov energy reaches zero again, it means that the state has reached the new commanded state. The time constant of the exponential energy decay is about 0.14 seconds, which is again longer than the expected value of 0.1 seconds, due to the slew rate limiting of the elevator deflections.

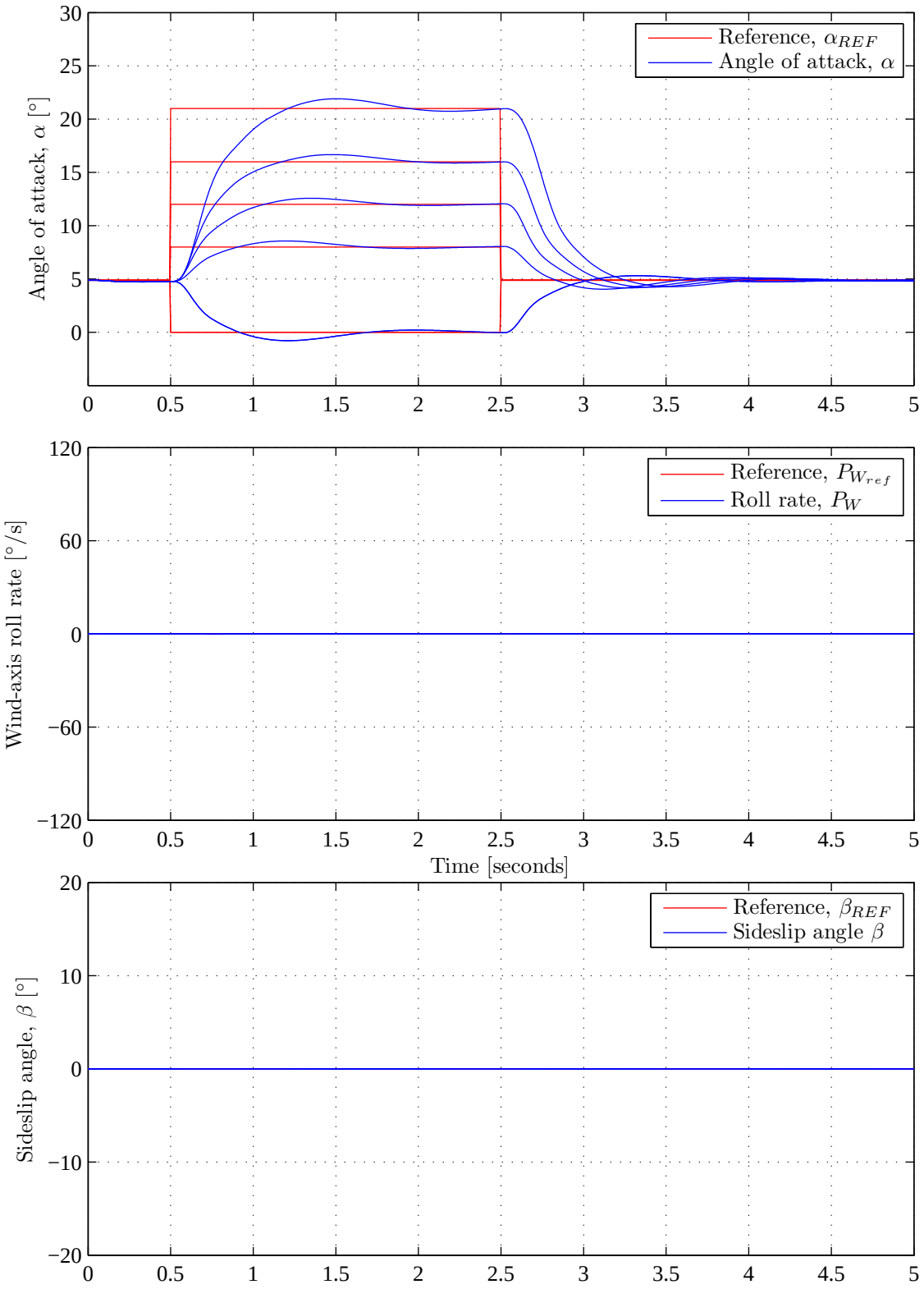


Figure 6.21: Angle of attack command tracking: angle of attack, sideslip angle, and wind-axis roll rate vs. time

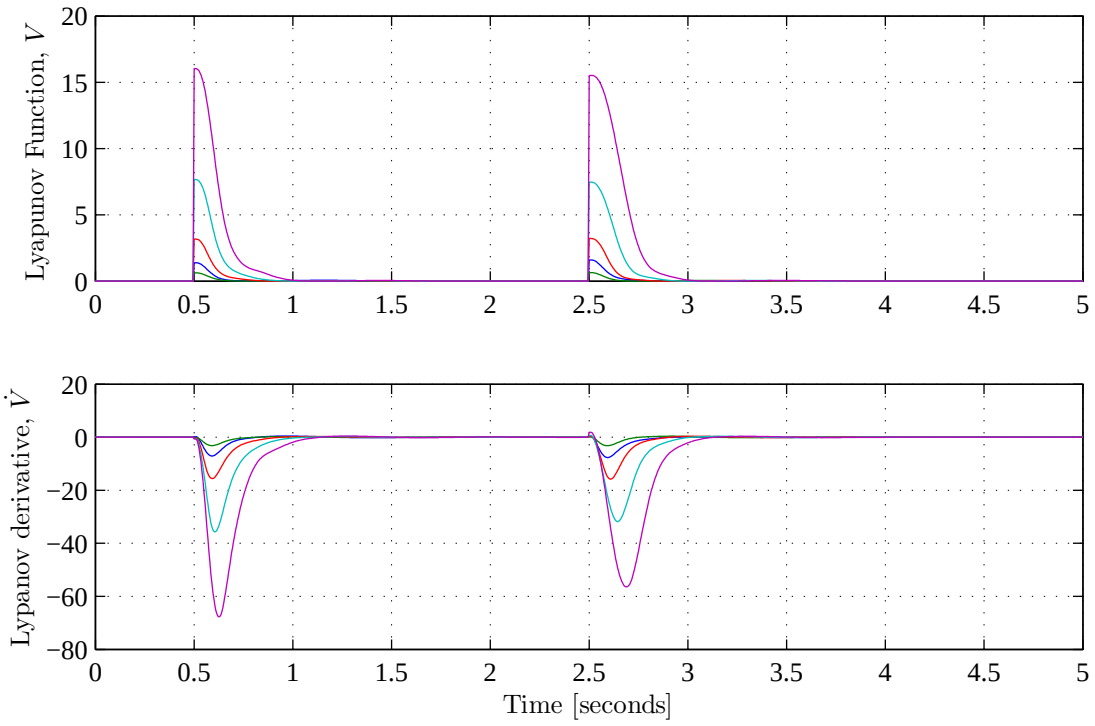


Figure 6.22: Angle of attack command tracking: Lyapunov energy and energy rate vs. time

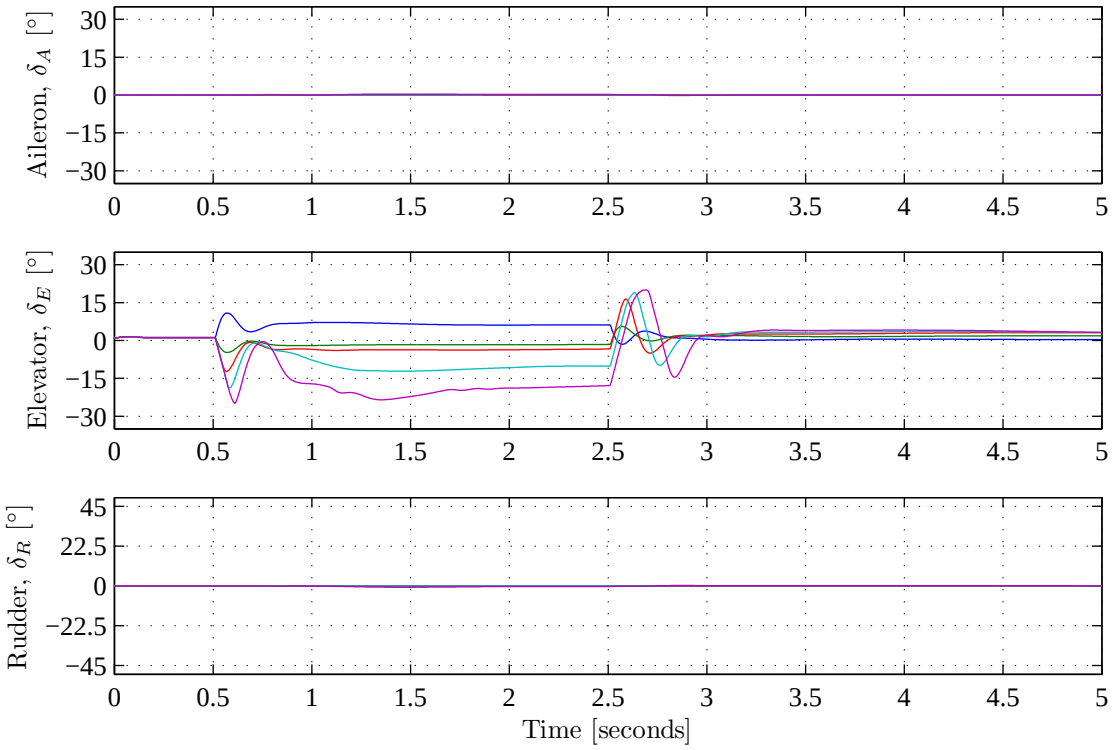


Figure 6.23: Angle of attack command tracking: control surface deflections vs. time

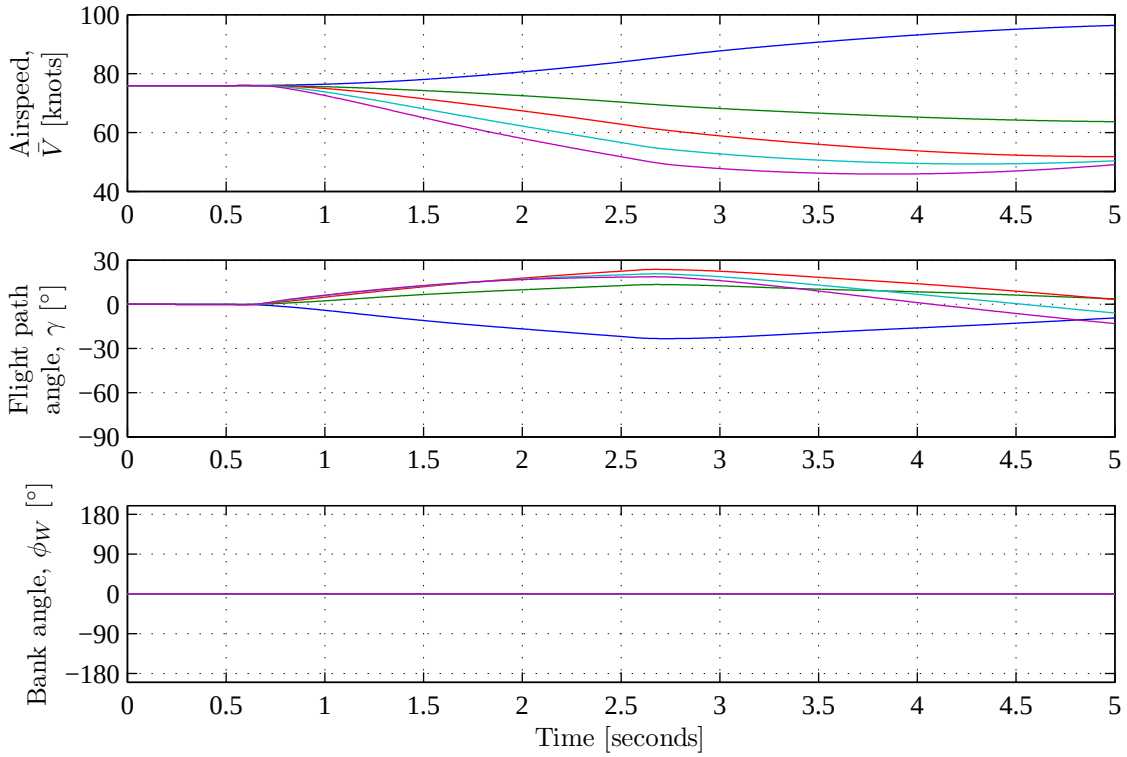


Figure 6.24: Angle of attack command tracking: airspeed, flight path angle, and bank angle vs. time

The control surface responses are shown in figure 6.23. The actuator responses show that the elevators are used to perform the angle of attack command tracking, and that there is very little usage of the ailerons and the rudders. Slew rate limiting is observed in the elevator response. In steady state, the elevators settle to a different trim deflection to maintain the angle of attack step, and then return to the original trim deflection to maintain the normal angle of attack of 4 degrees at the given airspeeds. Very small transients are observed in the aileron and rudder deflections, as they work to reject the small transient disturbances in the sideslip angle and body angular rates.

The time histories of the airspeed, flight path angle and bank angle are shown in figure 6.24, to show the translation motion of the aircraft centre of mass while the angle of attack command tracking is performed by the Lyapunov controller. For the first 0.5 seconds, the airspeed remains constant, the flight path angle remains level, and the bank angle remains zero. When the angle of attack is stepped to a higher value after t = 0.5 seconds, a positive flight path angle rate is observed due to the increased normal load factor produced by the increased lift associated with the higher angle of attack. The airspeed also decreases under the influence of gravity due to the increased drag and the positive flight path angle. (When the angle of attack is stepped to zero degrees, a negative flight path angle rate is observed due to the decreased normal load factor produced by the decreased lift associated with the lower angle of attack. The airspeed also increases under the influence of gravity due to the reduced drag and the negative flight path angle.) When the angle of attack is returned to 4 degrees after t = 2.5 seconds, a negative flight path angle rate is observed due to the decreased normal load factor associated with the low angle of attack. The airspeed also stops decreasing and starts increasing again under the influence of gravity as the flight path angle decreases and becomes negative. The bank angle is hardly disturbed during the maneuver and remains close to wings level due to the fact that the wind-axis roll rate is regulated to remain close to zero.

Angle of Attack and Roll Rate Command Tracking

The simulation results for simultaneous angle of attack and wind-axis roll rate command tracking are shown in figures 6.21 to 6.24. The results verify that the Lyapunov controller is able to accurately control the angle of attack and wind-axis roll rate of the aircraft in an extended aerodynamic envelope. (The Lyapunov controller's ability to simultaneously control the angle of attack and the wind-axis roll rate will be exploited in chapter 7 to recover the attitude, flight path angle, and airspeed of the aircraft.)

The aircraft is initialised with all flight variables inside the flight envelope (trim angle of attack, zero sideslip angle, zero angular rates, zero bank angle, zero flight path angle, and typical airspeed). The angle of attack reference for the Lyapunov controller is initially set to 4 degrees, and the sideslip angle and wind-axis roll rate references are both initially set to zero. At t = 0.5 seconds, the angle of attack reference is stepped to values of 0, 8, 12, 16 and 21 degrees, and the wind-axis roll rate reference is stepped to values of 0, ±30, ±60, and ±90 degrees per second. At t = 2.5 seconds, the angle of attack reference and the wind-axis roll rate reference are both stepped back to their initial values.

The angle of attack, sideslip angle, and wind-axis roll rate responses are shown in figure 6.21. The responses show that the Lyapunov controller is able to simultaneously track high angle of attack commands up to 21 degrees and high wind-axis roll rate commands up to 90 degrees per second. The transient responses look good with little overshoot and a fast settling time of about 1.5 seconds. The steady-state tracking performance also looks good with small tracking errors in both angle of attack and wind-axis roll rate. The sideslip angle is kept close to zero during the maneuver.

The control surface responses are shown in figure 6.26. Interestingly, the control surface deflections do not saturate during the simultaneous angle of attack and wind-axis roll rate steps at t = 0.5 seconds, and only the ailerons saturate when the references are stepped back at t = 2.5 seconds. The actuator responses indicate that the elevators are used to perform the angle of attack steps, and that the ailerons are used to perform the wind-axis roll rate steps. The rudder is mostly used to regulate the sideslip angle to zero, and it can be seen that the rudder usage is significantly less than the aileron and elevator usage. Slew rate limiting is observed in the elevator deflections at both t = 0.5 and t = 2.5 seconds.

The Lyapunov energy and Lyapunov energy rate responses are shown in figure 6.26. The responses show that the Lyapunov energy is always positive and that there are steps in the Lyapunov energy corresponding the simultaneous steps in the angle of attack reference and the wind-axis roll rate reference. Following each step, the Lyapunov energy decays back to zero with a response that resembles an exponential response. The time constant of the exponential energy decay ranges from 0.1 to 0.15 seconds, and is sometimes longer than the expected 0.1 seconds due to slew rate limiting and actuator saturation. The Lyapunov energy rate is negative most of the time, but does sometimes become slightly positive. The Lyapunov energy rate may become positive when the actuators saturate or when the wind axis angular rate references change rapidly due to normal and lateral translational accelerations experienced by the aircraft. Nonetheless, the simulation results show that the controller is stable for all command tracking trajectories, including those for which the Lyapunov energy rate becomes positive in the transient. As before, the Lyapunov controller provides exponential stability when the actuators do not saturate, and attempts to provide asymptotic stability when the actuators do saturate.

The airspeed, flight path angle, and bank angle responses are shown in figure 6.28. The responses confirm what we expect, which is that the flight path angle is controlled through the angle of attack, and the wind-axis bank angle is controlled through the wind-axis roll rate. The flight path angle increases at a constant rate for increased angles of attack, and decreases at a constant rate for reduced angles of attack.

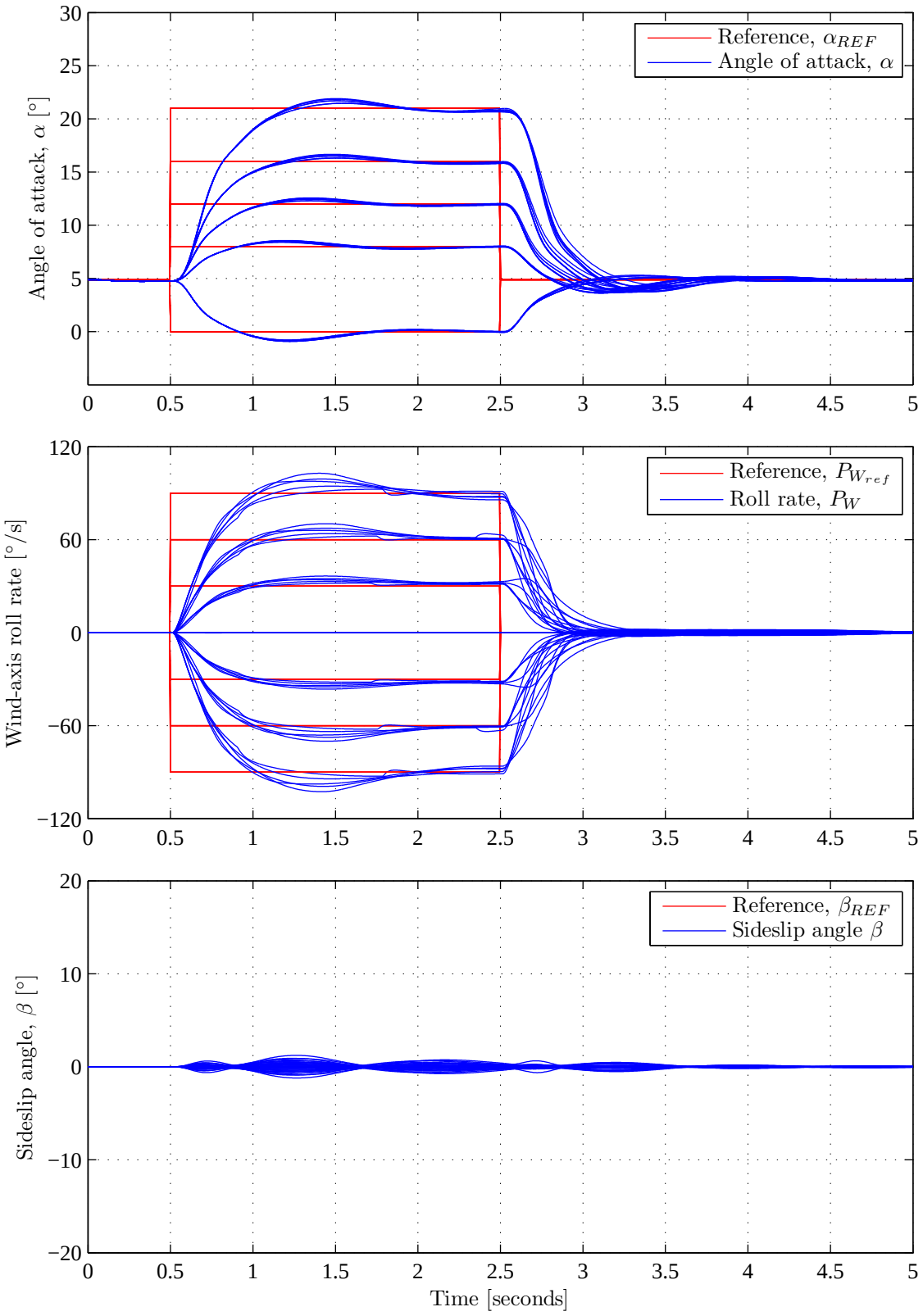


Figure 6.25: Angle of attack and roll rate command tracking: angle of attack, wind-axis roll rate, and sideslip angle vs. time

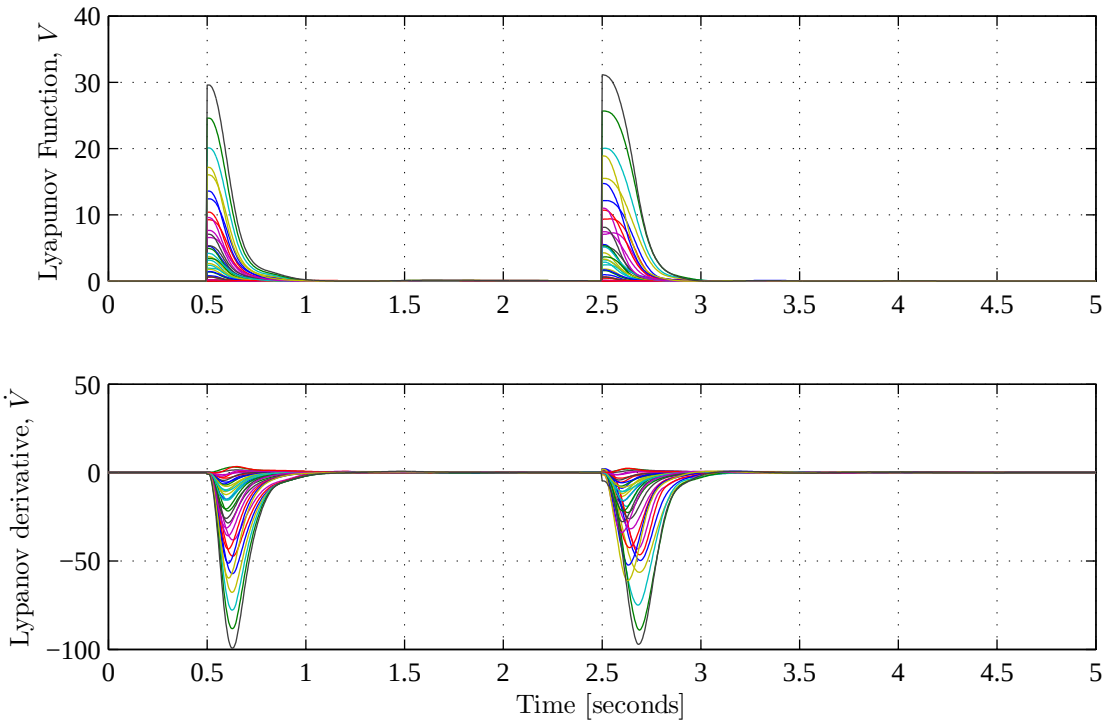


Figure 6.26: Angle of attack and roll rate command tracking: Lyapunov energy and energy rate vs. time

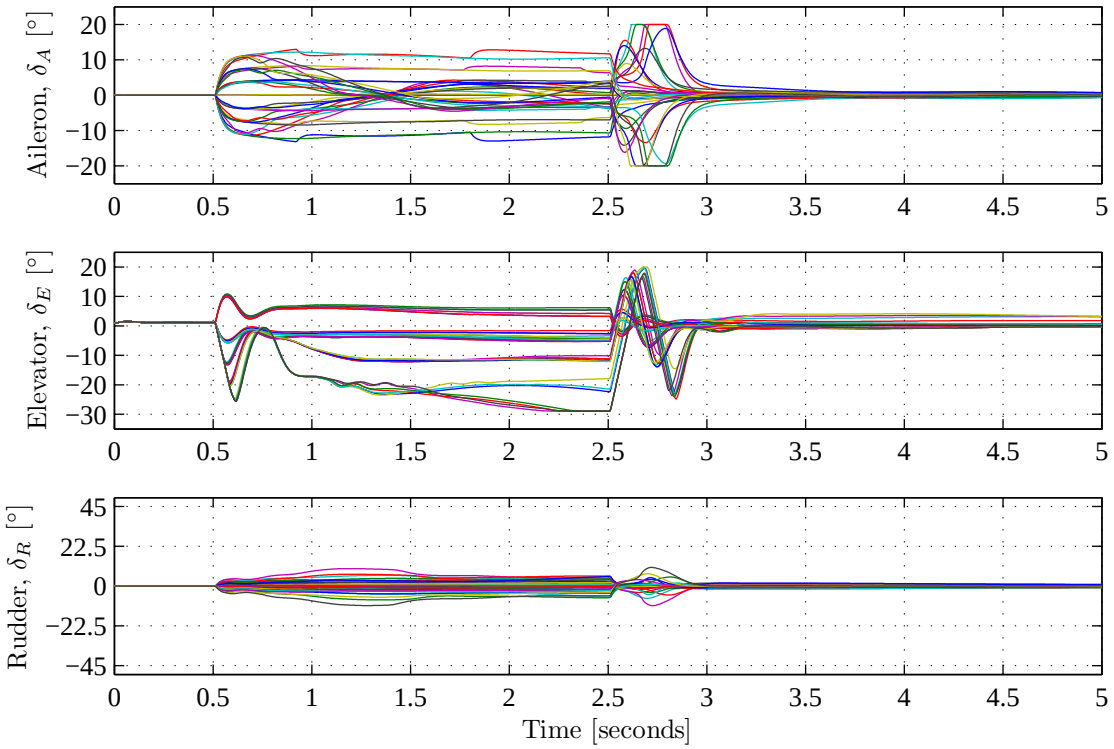


Figure 6.27: Angle of attack and roll rate command tracking: control surface deflections vs. time

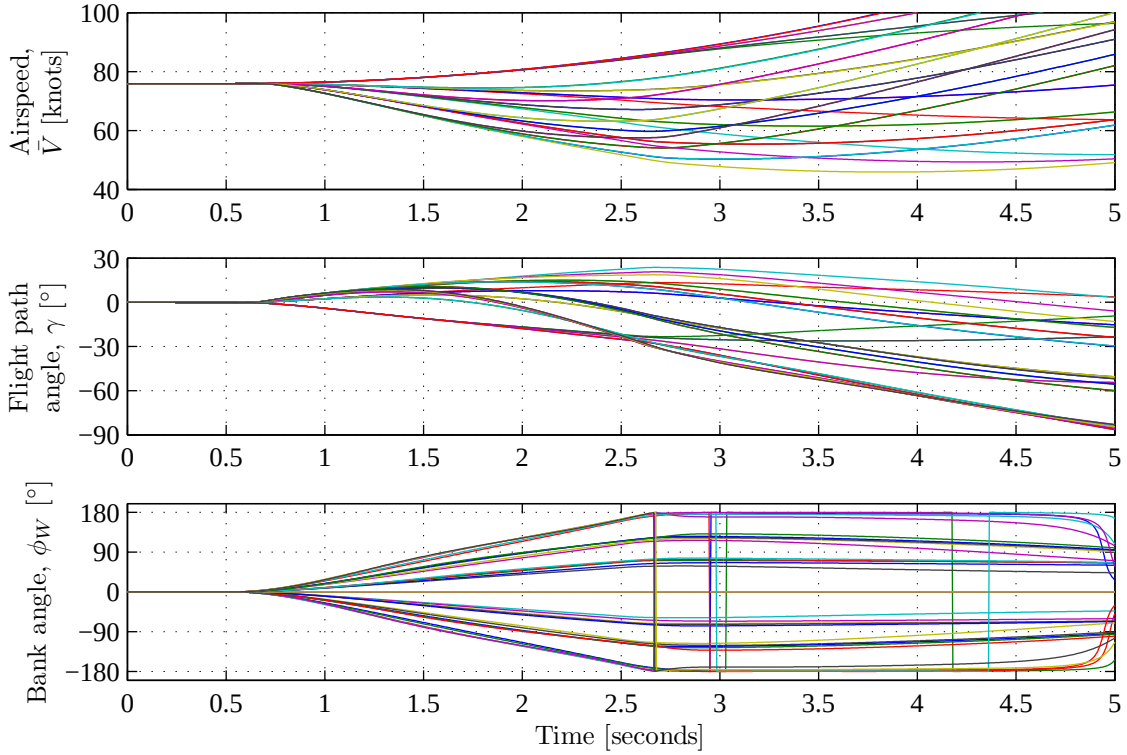


Figure 6.28: Angle of attack and roll rate command tracking: airspeed, flight path angle, and bank angle vs. time

Similarly, the bank angle increases at a constant rate for positive roll rates, and decreases at a constant rate for negative roll rates. When the wind-axis roll rate reference is set to zero after t = 2.5 seconds, the rolling motion is arrested and the wind-axis bank angle remains constant.

The airspeed is affected by a combination of the angle of attack and the flight path angle. For higher angles of attack, the induced drag is higher and the airspeed tends to decrease over time. For lower angles of attack, the induced drag is lower, and the airspeed tends to increase over time. For positive flight path angles, the gravitational force tends to decrease the airspeed over time, while for negative flight path angles, the gravitational force tends to increase the airspeed over time.

Angle of Attack, Roll Rate and Sideslip Angle Tracking

The simulation results for the command tracking of simultaneous angle of attack, wind-axis roll rate *and* non-zero sideslip angle references are shown in figures 6.29 to 6.32. This represents a very unusual and challenging maneuver for the aircraft, and the simulation tests are mainly performed to push the limits of Lyapunov controller. The results show that the Lyapunov controller succeeds in simultaneously controlling the angle of attack, wind-axis roll rate, and sideslip angle to follow their respective reference commands, but struggles to regulate the tracking errors to zero for high sideslip angle references.

The aircraft is initialised with all flight variables inside the flight envelope (trim angle of attack, zero sideslip angle, zero angular rates, zero bank angle, zero flight path angle, and typical airspeed). The angle of attack reference for the Lyapunov controller is initially set to 4 degrees, and the sideslip angle and wind-axis roll rate references are both initially set to zero.

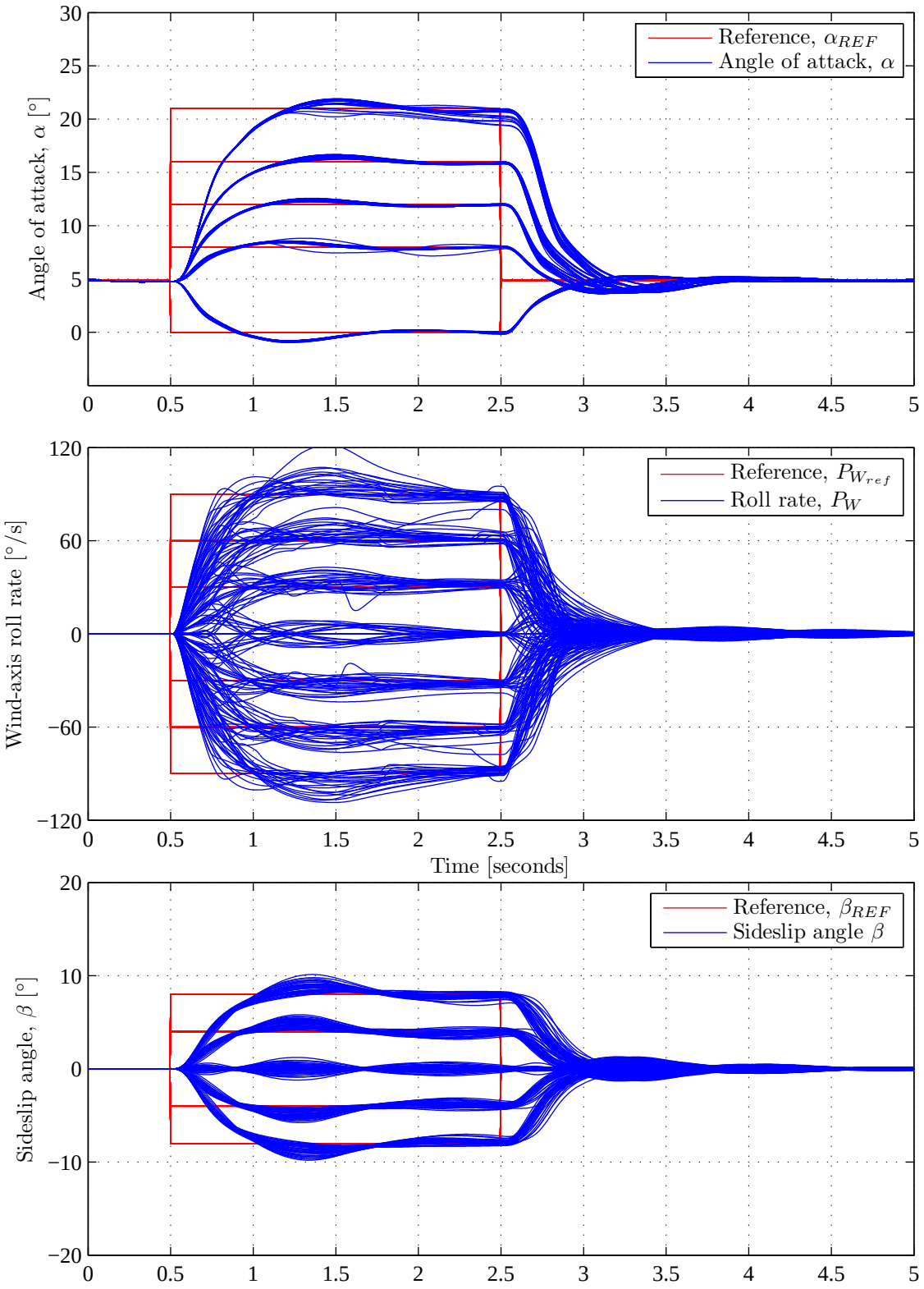


Figure 6.29: Angle of attack, roll rate, and sideslip angle command tracking: angle of attack, wind-axis roll rate, and sideslip angle vs. time

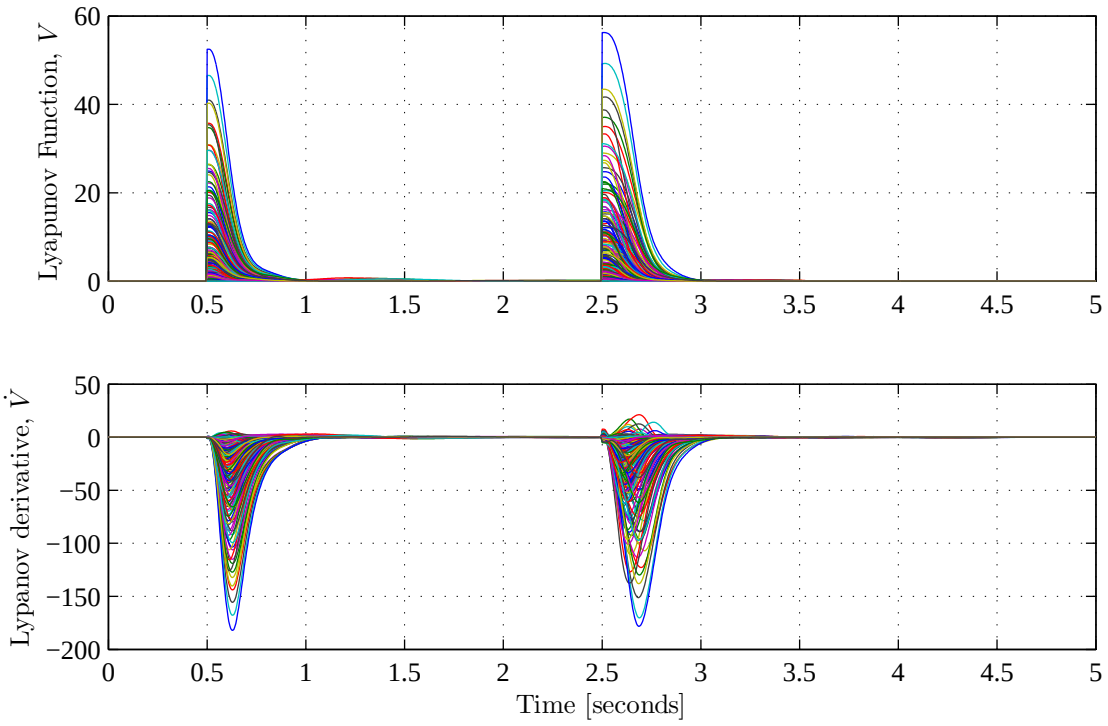


Figure 6.30: Angle of attack, roll rate, and sideslip angle command tracking: Lyapunov energy and energy rate vs. time

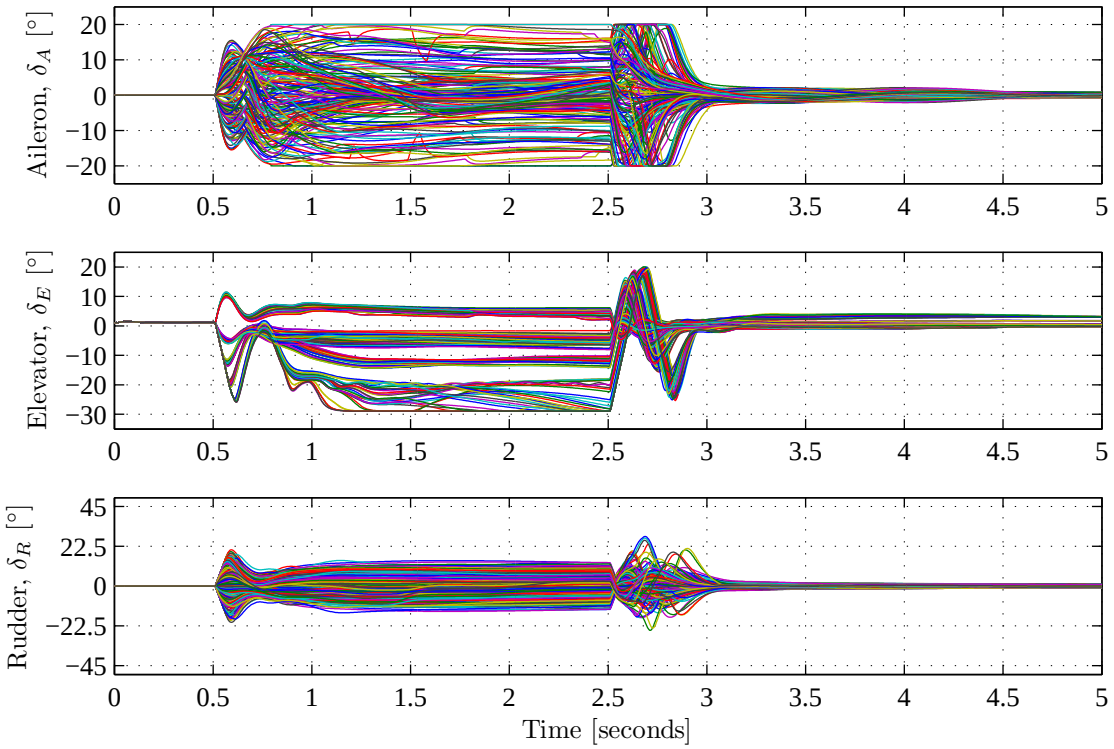


Figure 6.31: Angle of attack, roll rate, and sideslip angle command tracking: control surface deflections vs. time

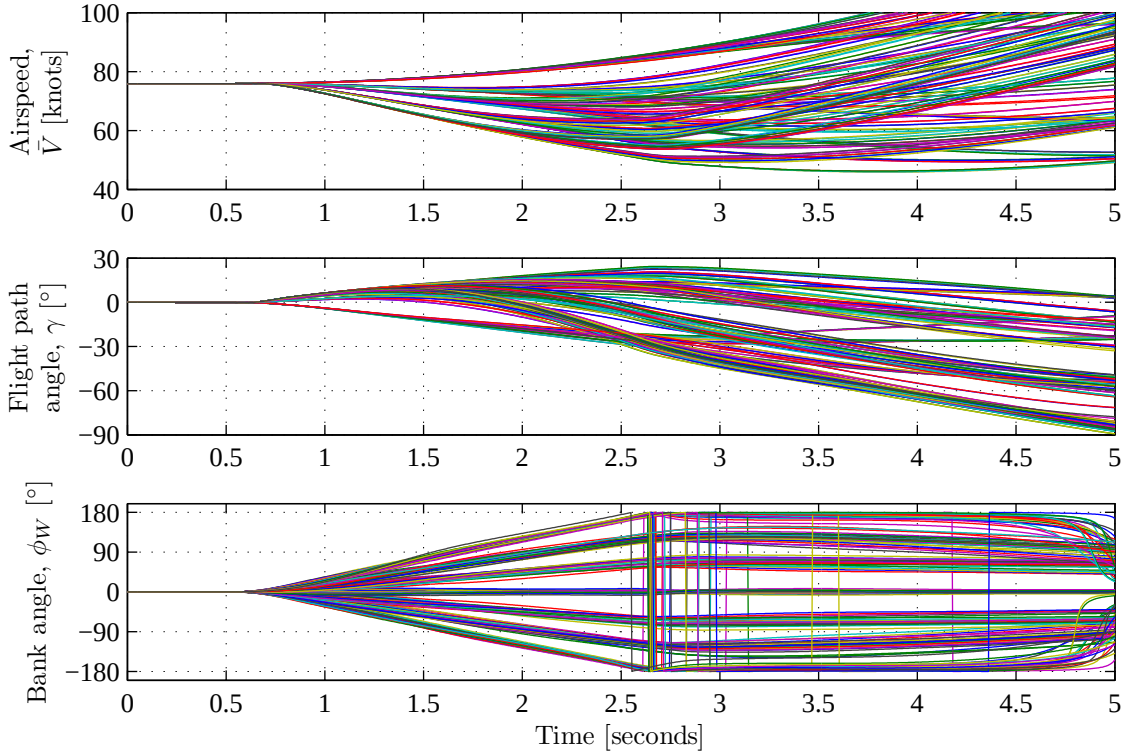


Figure 6.32: Angle of attack, roll rate, and sideslip angle command tracking: airspeed, flight path angle, and bank angle vs. time

At t = 0.5 seconds, the angle of attack reference is stepped to values of 0, 8, 12, 16 and 21 degrees, the wind-axis roll rate reference is stepped to values of 0, ±30, ±60, and ±90 degrees per second, and the sideslip angle reference is stepped to values of 0, ±4 and ±8 degrees. At t = 2.5 seconds, all three references are stepped back to their initial values.

The angle of attack, sideslip angle, and wind-axis roll rate responses are shown in figure 6.29. The responses show that the Lyapunov controller is able to simultaneously track high angle of attack commands up to 21 degrees and high wind-axis roll rate commands up to ±90 degrees per second, and sideslip angle references up to ±8 degrees. The transient responses are similar to the responses achieved when only angle of attack steps and wind-axis roll rate steps are commanded, except that more variations are observed in the wind-axis roll rate response for high sideslip angle references.

The control surface responses are shown in figure 6.30. After the step at t = 0.5 seconds, and for combined high angle of attack, high wind-axis roll rate and high sideslip angle reference commands, the ailerons and the elevators both saturate. The ailerons saturate as they attempt to maintain the commanded wind-axis roll rate while countering the large rolling moment produced by the sustained high sideslip angle. The elevators saturate as they attempt to maintain the commanded angle of attack. Although the rudder does not saturate, significantly more rudder is used to maintain the commanded non-zero sideslip angle. After the reference commands are stepped back to their normal values at t = 2.5 seconds, the actuators return to their normal trim values corresponding to the resulting airspeeds experienced by the aircraft after the maneuver.

The Lyapunov energy and Lyapunov energy rate responses are shown in figure 6.30. The energy and energy rate responses are similar to the responses achieved when only angle of attack steps and wind-axis roll rate steps are commanded. The Lyapunov energy is always positive, Lyapunov energy steps are observed corresponding to the simultanous steps in angle of attack,

wind axis-roll rate and sideslip angle references, and following each step the Lyapunov energy decays back to zero with an exponential-like response with a time constant between 0.1 and 0.15 seconds. The Lyapunov energy rate is negative most of the time, but does sometimes become slightly positive due to actuator saturation and rapidly changing wind-axis angular rates. Nonetheless, the simulation results show that the controller is stable for all command tracking trajectories, including those for which the Lyapunov energy rate becomes positive in the transient. As before, the Lyapunov controller provides exponential stability when the actuators do not saturate, and attempts to provide asymptotic stability when the actuators do saturate.

The airspeed, flight path angle, and bank angle responses are shown in figure 6.32 and are affected in the same way as for the cases where only angle of attack and wind-axis roll rate steps are given, and the sideslip angle reference is kept at zero.

6.7 Conclusions

The simulation results on the full NASA GTM simulation model show that the Lyapunov controller is able to actively recover the angular rates and aerodynamic envelope of the aircraft from out-of-envelope conditions. The Lyapunov controller successfully performs active angle of attack recovery, active sidesip angle recovery, and active angular rate recovery from initial out-of-envelope states with angles of attack from -5 to +60 degrees, sideslip angles from -45 to +45 degrees, and wind-axis angular rates from -180 to +180 degrees per second. For all initial states, the angle of attack, the sideslip angle and the body angular rates were successfully recovered to desired references within the aerodynamic envelope. The transient response exhibited good damping and a good speed of response with a settling time of 0.1 to 0.15 seconds. In steady state, the angle of attack, sideslip angle, and wind-axis roll rate tracked their respective references with zero errors.

The simulation results also show that the Lyapunov controller is able to actively control the angle of attack, wind-axis roll rate and sideslip angle to track reference commands in an extended aerodynamic envelope. The Lyapunov controller is able to simultaneously and accurately control the angle of attack to track commands from 0 to 21 degrees, the wind-axis roll rate to track commands from -90 to +90 degrees per second, and the sideslip angle to track commands from -8 to +8 degrees. The command tracking ability of the Lyapunov controller will be used in the next chapter when recovering the attitude, flight path angle and airspeed of the aircraft. The Lyapunov controller will be used as an inner-loop controller for an optimal attitude and flight vector recovery guidance law.

6.8 Summary of Contributions

The following contributions were made in this chapter:

- The active aerodynamic envelope recovery was conceptualised as the problem of controlling the fast rotational dynamics of the aircraft to recover the angle of attack, sideslip angle and angular rates of the aircraft from out-of-envelope states, and to regulate them towards their desired equilibrium states inside the normal aerodynamic envelope.

- The aerodynamic envelope recovery was decoupled from the translational motion of the aircraft, by arguing that the rotational motion of the body axis system relative to the wind axis system operates over much shorter time scales than the rotational motion of the wind axis system relative to the inertial axis system. This allowed the plant to be formulated as the reduced-order model of the aircraft's fast rotational dynamics.

- A novel Lyapunov-based inner-loop controller was developed to actively control the angle of attack, sideslip angle, and angular rates of the aircraft using the ailerons, elevators, and rudder. The Lyapunov controller is formulated to be general and can be applied to any aircraft for which a wide-envelope aerodynamic model is available, and that has the standard set of air data sensors and inertial sensors available on a typical large transport aircraft.

- A novel Lyapunov function was proposed that represents the state of the fast rotational dynamics of the aircraft as an energy-like function. Lyapunov-based control laws were then derived to control the time derivative of the energy to always be negative and proportional to the energy, thus providing exponential Lyapunov stability.

- A method for the solving the actuator deflections to produce the angular accelerations required by the Lyapunov control law was proposed. The actuator deflections are solved using a multi-variable secant method that takes actuator saturation into account. When the actuators do not saturate, the Lyapunov controller provides exponential stability. When the actuators saturate, the Lyapunov controller still attempts to provide at least asymptotic stability.

- The Lyapunov-based controller implicitly provides gain scheduling for different airspeeds and altitudes, and also implicitly handles inertial coupling. The Lyapunov controller should display significant robustness to model uncertainty due to the nature of Lyapunov-based control and because the proposed architecture relies mostly on kinematic models that contain very little uncertainty.

- A novel method for sensing the rotational motion of the wind axis system relative to the inertial axis system using the available onboard inertial sensors was introduced. The sensed angular rates of the wind axis system are supplied as references to the Lyapunov-based controller to help the body axis system track the wind axis system better.

- Extensive simulation tests were performed on the NASA Generic Transport Model to verify the Lyapunov-based controller's ability to perform high angular rate recovery and aerodynamic envelope recovery. The simulation results show that the Lyapunov controller is able to recover the aircraft from initial states with angles of attack from -5 to +60 degrees, sideslip angles from -45 to +45 degrees, and wind-axis angular rates from -180 to +180 degrees per second.

- Extensive simulation tests were performed on the NASA Generic Transport Model to verify the Lyapunov-based controller's ability to perform command tracking for the angle of attack, wind-axis roll rate, and sideslip angle in an extended aerodynamic envelope. The simulation results show that the Lyapunov controller is able to simultaneously and accurately control the angle of attack to track commands from 0 to 21 degrees, the wind-axis roll rate to track commands from -90 to +90 degrees per second, and the sideslip angle to track commands from -8 to +8 degrees.

Chapter 7

Optimal Attitude and Flight Vector Recovery

In this chapter, an optimal attitude and flight vector recovery guidance law is proposed that simultaneously recovers the bank angle, the flight path angle, and the airspeed of the aircraft from upset conditions while minimising the total altitude lost during the recovery maneuver. The optimal attitude and flight vector recovery guidance law is an outer-loop guidance law that uses the Lyapunov-based controller from chapter 6 as an inner-loop controller. The inner-loop Lyapunov-based controller performs the high angular rate and aerodynamic envelope recovery, and the outer-loop guidance law performs the attitude envelope recovery and the flight vector envelope recovery, while remaining within the structural integrity envelope.

The recovery problem is formulated as an optimal control problem and then solved using a dynamic programming algorithm to find the optimal state trajectories and the optimal sequence of control inputs. A novel hierarchical multi-objective cost function is introduced. A primary cost function is defined to minimise the altitude lost, a secondary cost function is defined to minimise the maximum airspeed reached, and a tertiary cost function is defined to recover the bank angle to wings level as soon as possible. The aircraft dynamics is simplified to the reduced-order model describing the slower point mass translational dynamics of the aircraft, while the fast rotational dynamics is abstracted through time scale separation. The reduced-order model of the aircraft dynamics makes the optimal control problem tractable to be solved with dynamic programming.

The dynamic programming approach has several advantages. Firstly, the dynamic programming algorithm produces a lookup table of the optimal state trajectories and optimal control sequences for recovery from *all* possible initial states. Secondly, the lookup table explicitly stores the altitude that will be lost during optimal recovery for all states. Given the state of the aircraft, the lookup table therefore predicts how much altitude would be lost if the recovery procedure is activated immediately. Thirdly, although the dynamic programming search algorithm is computationally intensive, it produces a lookup table that is computationally light in comparison.

The optimal attitude and flight vector recovery guidance law is verified in simulation using the NASA Generic Transport Model. The simulation results show that the guidance law is able to successfully perform combinations of flight path angle recovery (steep descent, steep climb), bank angle recovery (high and inverted bank angles), underspeed recovery, and overspeed recovery while obeying the constraints of the structural integrity envelope. The comprehensive set of optimal state trajectories and control inputs from all recoverable states are presented, along with the load factor trajectories and altitude trajectories. Maps of the recoverable and unrecoverable initial states are also shown. Finally, the optimal recovery trajectories and control inputs for some example recovery scenarios are presented and discussed.

7.1 Objectives

The objective of the optimal attitude and flight vector recovery is to recover the bank angle to wings level, the flight path angle to level flight, and the airspeed to an acceptable range, with the minimum of altitude loss, while adhering to the constraints of the aircraft dynamics and structural limitations.

The attitude and flight vector recovery therefore represents the recovery of the attitude envelope (bank angle and pitch angle) and the recovery of the flight vector envelope (airspeed magnitude and flight path angle) while adhering to the structural integrity envelope (normal load factor limits and maximum airspeed limit). The attitude and flight vector recovery assumes that the aerodynamic envelope has already been recovered by an inner-loop controller, such as the Lyapunov-based controller.

The bank angle, flight path angle, and airspeed are recovered explicitly, while the pitch angle is recovered implicitly. If the bank angle is zero, then the pitch angle equals the sum of the flight path angle and the angle of attack. If the aerodynamic envelope has been recovered, then the angle of attack will be quite small, and the pitch angle will be approximately equal to the flight path angle. If the angle of attack has already been recovered, then recovering the flight path angle implicitly also recovers the pitch angle.

7.2 Approach

The attitude and flight vector recovery problem is formulated as an optimal control problem and solved using a dynamic programming algorithm to find the optimal state trajectories and optimal control input sequences to minimise the peak altitude lost during the recovery maneuver while adhering to the constraints of the aircraft dynamics and structural limitations.

We argue that the problem can be reduced to consider only the translational dynamics of the aircraft by exploiting the time scale separation between the rigid body rotational dynamics and the point mass translational dynamics of the aircraft. The rigid body rotational dynamics describes the rotational motion of the aircraft body relative to its own velocity vector, primarily under the influence of the aerodynamic moments, and include states such as angle of attack, sideslip angle and body angular rates, and inputs such as aileron, elevator and rudder deflections. The point mass translational dynamics describes the dynamics of the aircraft velocity vector relative to the inertial reference frame, under the influence of the aerodynamic, gravitational and engine forces, and include states such as aircraft velocity, flight path angle, and wind-axis bank angle, and inputs such as angle of attack, wind-axis roll rate, and engine thrust. The fast rotational dynamics is encapsulated by the inner-loop Lyapunov controller and changes over much shorter time scales than the point mass translational dynamics. The time scale separation may therefore be exploited to neglect the fast rotational dynamics and reduce the model to only the point mass translational dynamics. The assumption is therefore made that the angle of attack, sideslip angle, and wind-axis roll rate instantaneously follow their respective commands. The reduced-order formulation of the system dynamics makes the optimal control problem tractable to be solved with dynamic programming.

The attitude and flight vector recovery problem is formulated as an optimal control problem as follows: The objective of minimising the altitude loss is represented by a cost function. A novel hierarchical multi-objective cost function is introduced to represent the primary objective to minimise the altitude lost, a secondary objective to minimise the maximum airspeed reached, and a tertiary objective to recover the bank angle to wings level as soon as possible. Sets of admissible states are defined to constrain the ranges of allowable airspeed, flight path angle, and bank angle. Sets of admissible control inputs are defined to constrain the ranges of allowable angle of attack command, wind-axis roll rate command, and thrust command. The problems of underspeed and overspeed recovery are addressed by including underspeed and overspeed states

in the range of admissible states, but only including acceptable speeds in the set of admissible *final* states. The load factor constraints are represented by constraining the allowable angle of attack input as a function of the airspeed state. Aerodynamic envelope constraints are represented by constraining the wind-axis roll rate input as a function of the angle of attack input.

Two approaches for dealing with significant engine lag dynamics are proposed. In the first approach, the engine thrust is not used as a control input, and the thrust command is kept constant. In the second approach, the thrust is treated as a state variable and the thrust command as a control input. The thrust variable increases the dimension of the state vector by one, and makes the problem more computationally challenging for a dynamic programming approach.

The optimal control problem is solved using a dynamic programming algorithm. Dynamic programming is a search algorithm that models an optimal control problem as a multi-stage decision process. The dynamic programming algorithm produces a lookup table that is used as an outer-loop guidance law to generate the optimal state trajectories and control input sequences from a given initial state. The optimal control input sequences may be provided as reference signals for the inner-loop Lyapunov-based controller. Alternatively, the optimal state trajectories may be provided as reference signals for middle-loop feedback control laws that control the airspeed, the flight path angle and the bank angle, and the optimal control input sequences may be provided as feedforward reference signals for inner-loop feedback control laws that control the angle of attack, the wind-axis roll rate, and the thrust.

7.3 Point Mass Translational Dynamics

The reduced-order model of the aircraft's point mass translational dynamics serves as the basis for the optimal attitude and flight vector recovery guidance law. The aircraft dynamics is simplified by considering only the translational motion of the aircraft as a point mass under the influence of the aerodynamic, engine, and gravitational forces. The rotational motion of the aircraft is abstracted away through time scale separation, and it is assumed that the angle of attack, wind-axis roll rate, and the sideslip angle change instantaneously from the perspective of the translational motion. The point mass translational motion can also be thought of as the motion of the wind axis system relative to the inertial axis system, while abstracting away the motion of the body axis system relative to the wind axis system.

The velocity vector of the aircraft point mass relative to the inertial axis system may be expressed in terms of the velocity magnitude $\overline{V}$, the flight path angle γ, and the flight path heading ψ_W as shown in figure 7.1. The rotation of the aircraft about its own velocity vector may be expressed in terms of the wind-axis bank angle ϕ_W. The forces acting on the aircraft point mass are the aerodynamic lift, the aerodynamic drag, the aerodynamic side force, the engine thrust, and the gravitational force. The aerodynamic lift vector acts perpendicular to the aircraft velocity vector and the aerodynamic drag vector acts parallel to the aircraft velocity vector, but in the opposite direction. (The aerodynamic side force vector acts perpendicular to both the aircraft velocity vector and the lift vector.) The lift vector is rotated about the aircraft velocity vector through the wind-axis bank angle. The engine thrust vector acts approximately parallel to the aircraft velocity vector, and in the same direction. The gravitational force vector acts in the inertial down direction.

The aerodynamic lift force, drag force, and side force are controlled through the angle of attack and the sideslip angle, as determined by the nonlinear wide-envelope aerodynamic model of the aircraft. The direction of the lift vector is determined by the wind-axis bank angle and is controlled through the wind-axis roll rate. The thrust vector is controlled through the engine thrust command. The engine lag dynamics may be modelled as a first-order response.

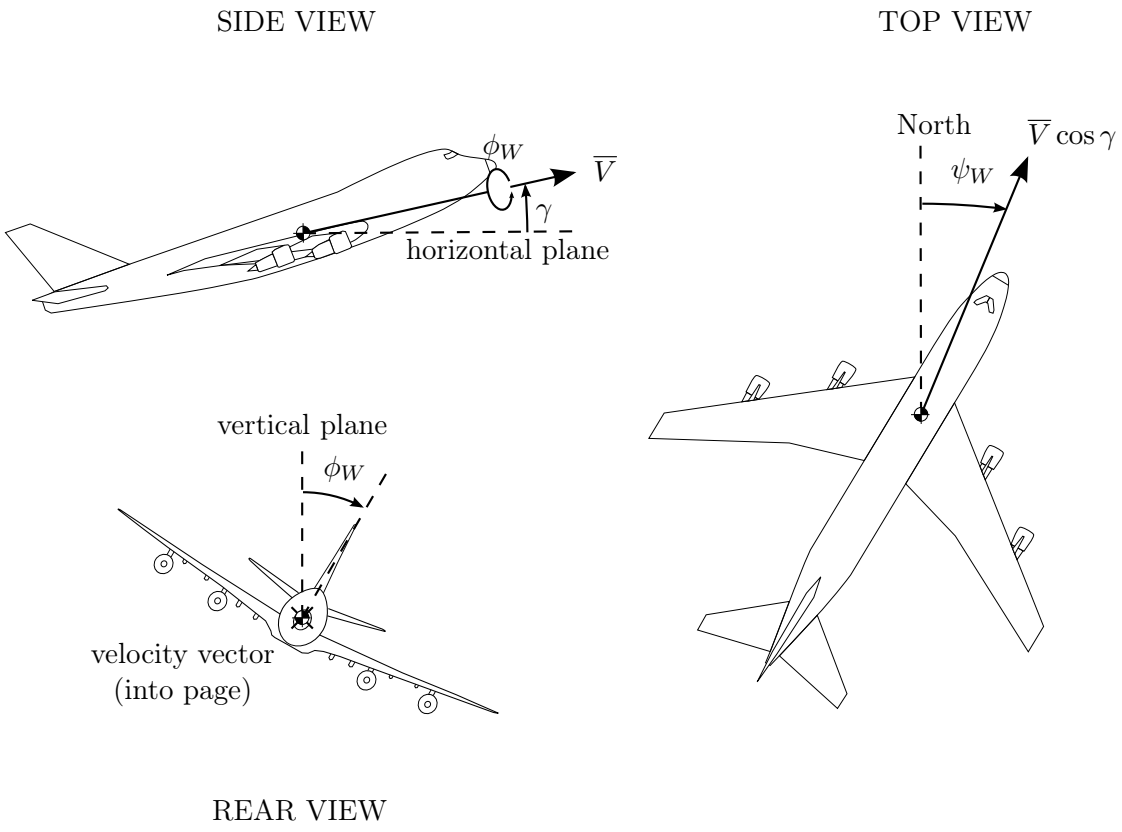


Figure 7.1: Visualisations of airspeed magnitude $\overline{V}$, flight path angle γ, flight path heading ψ_W, and wind-axis bank angle ϕ_W

The airspeed magnitude $\overline{V}$, the flight path angle γ, and the flight path heading ψ_W express the aircraft velocity vector using spherical coordinates. Alternatively, the aircraft velocity vector may be expressed using Cartesian coordinates in terms of north velocity V_N, east velocity V_E, and down velocity V_D. The spherical coordinates are related to the Cartesian coordinates through the following relationship

$$\overline{V} = \sqrt{V_N^2 + V_E^2 + V_D^2} \tag{7.1}$$

$$\gamma = \arctan\left(\frac{-V_D}{\sqrt{V_N^2 + V_E^2}}\right) \tag{7.2}$$

$$\psi_W = \arctan_2\left(\frac{V_E}{V_N}\right) \tag{7.3}$$

and the inverse relationship

$$V_N = \overline{V}\cos\gamma\cos\psi_W \tag{7.4}$$

$$V_E = \overline{V}\cos\gamma\sin\psi_W \tag{7.5}$$

$$V_D = -\overline{V}\sin\gamma \tag{7.6}$$

The relationship between the aircraft position and its velocity in Cartesian coordinates is ex-

pressed by the following simple set of differential equations

$$\begin{bmatrix} \dot{N} \\ \dot{E} \\ \dot{D} \end{bmatrix} = \begin{bmatrix} V_N \\ V_E \\ V_D \end{bmatrix} \tag{7.7}$$

The altitude h and altitude rate $\dot{h}$ are the negatives of the down position D and the down velocity V_D, respectively.

$$h = -D \tag{7.8}$$
$$\dot{h} = -V_D = \overline{V} \sin\gamma \tag{7.9}$$

During the recovery, we are going to control the roll angle of the aircraft to direct the lift vector. We therefore also need to include the wind-axis roll angle in the dynamic model of the point mass translational dynamics. The orientation of the aircraft's wind axis system relative to the inertial axis system can be expressed in terms of the flight path heading ψ_W, the flight path angle γ, and the wind axis bank angle ϕ_W and represented by an Euler 3-2-1 sequence of rotations as shown in figure 7.2. The sequence starts with the wind axis system aligned with the inertial axis system, and then rotates the wind axis system through the following set of ordered rotations:

1. Rotate the wind axis system about the inertial z-axis through the flight path heading ψ_W.
2. Rotate the wind axis system about the new y-axis through the flight path angle γ.
3. Rotate the wind axis system about the new x-axis through the wind-axis bank angle ϕ_W.

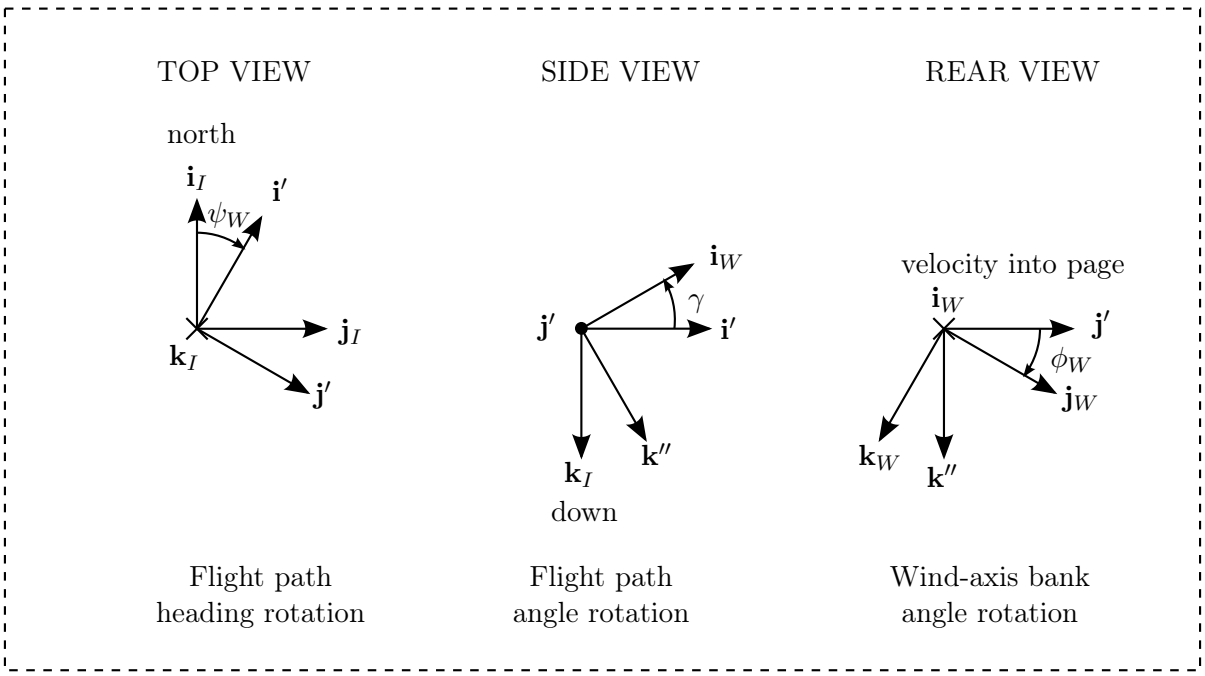


Figure 7.2: Euler 3-2-1 rotation from inertial axes to wind axes

The point mass translational dynamics may be derived with the aid of figure 7.3 and expressed

with the following set of differential equations

$$\dot{\overline{V}} = \frac{T - D - mg\sin\gamma}{m} \tag{7.10}$$

$$\dot{\gamma} = \frac{1}{\overline{V}} \frac{L\cos\phi_W - Y\sin\phi_W + mg\cos\gamma}{m} \tag{7.11}$$

$$\dot{\psi}_W = \frac{1}{\overline{V}\cos\gamma} \frac{L\sin\phi_W + Y\cos\phi_W}{m} \tag{7.12}$$

$$\dot{\phi}_W = P_W \tag{7.13}$$

where L, D, and Y are the aerodynamic lift force, drag force, and side force respectively, T is the engine thrust force, and P_W is the wind-axis roll rate.

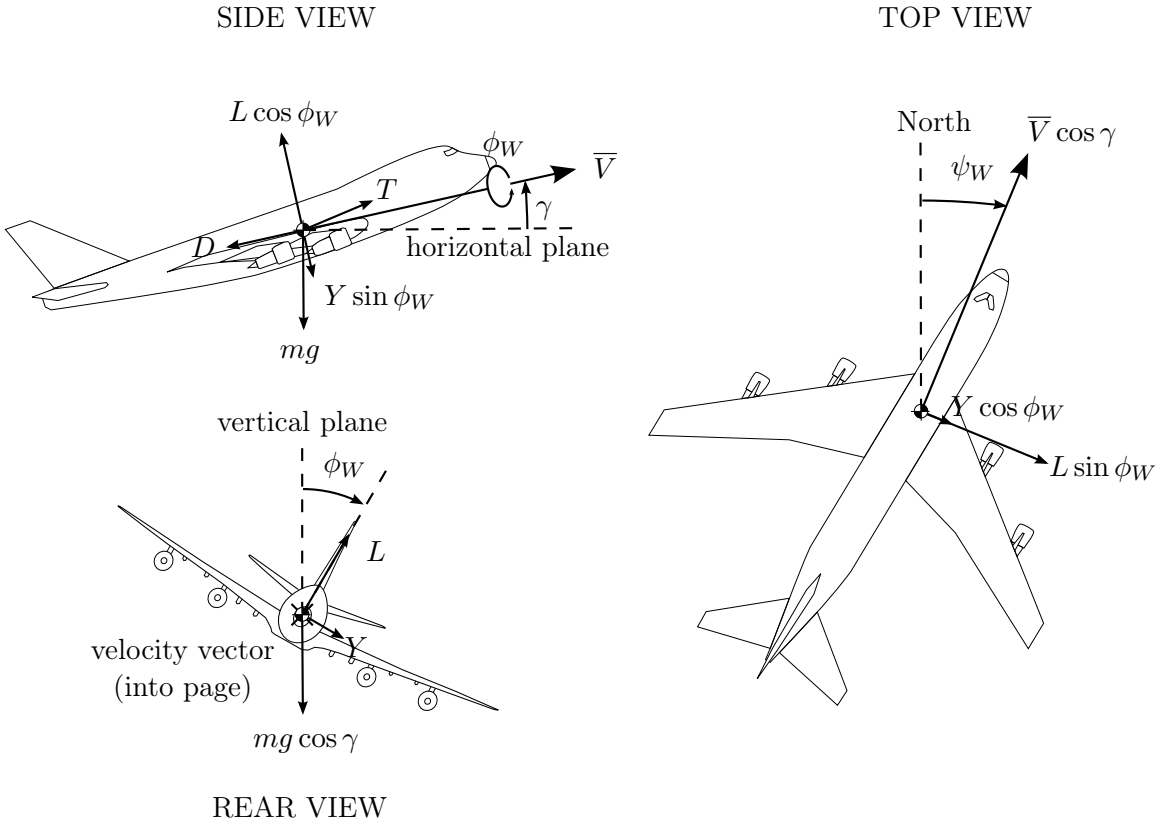


Figure 7.3: Point mass translational dynamics

From the equations above it can be seen that the velocity magnitude is affected by the engine thrust, the aerodynamic drag, and the component of the gravity in the wind axis axial direction. The flight path angle is affected by the component of aerodynamic lift in the vertical direction, the component of aerodynamic side force in the vertical direction, and the component of gravity in the vertical direction. The flight path heading is affected by the component of aerodynamic lift in the lateral direction, and the component of aerodynamic side force in the lateral direction. The wind axis roll angle is affected only by the wind-axis roll rate.

The aerodynamic lift, side force, and drag can be modelled with the following simplified equations

$$L = \frac{1}{2}\rho\overline{V}^2 SC_L(\alpha,\beta) \tag{7.14}$$

$$Y = \frac{1}{2}\rho\overline{V}^2 SC_Y(\alpha,\beta) \tag{7.15}$$

$$D = \frac{1}{2}\rho\overline{V}^2 SC_D(\alpha,\beta) \tag{7.16}$$

where ρ is the air density, S is the wing surface area, and C_L, C_Y, and C_D are the non-dimensional aerodynamic coefficients for lift, side force and drag respectively. The aerodynamic equations have been simplified by arguing that the lift, side force and drag are primarily produced by the dynamic pressure $\frac{1}{2}\rho\overline{V}^2$, the angle of attack α, and the sideslip angle β, and that the lift, side force, and drag contributions from the angular rates and the actuator deflections may be neglected.

The engine thrust vector has been modelled as if it only acts in the direction of the velocity vector and that there are no components of thrust perpendicular to the velocity vector. In reality, the orientation of the engine thrust relative to the aircraft body, as well as the orientation of the body axes relative to the wind axes in terms of angle of attack and sideslip angle, would result in the engine thrust having components in both the wind axis normal and the wind axis lateral directions. However, it can be argued that for large transport aircraft without thrust vectoring, the thrust vector should primarily be used to control the magnitude of the velocity vector, and should have very little direct effect on the flight path angle and flight path heading compared to the effects of the aerodynamic forces and gravity force.

The engine lag dynamics can also be included in the point mass translational dynamics with the following differential equation

$$\dot{T} = -\frac{1}{\tau}T + \frac{1}{\tau}T_c \tag{7.17}$$

where T is the engine thrust, T_c is the thrust command, and τ is the engine time constant. The lag dynamics of the angle of attack, sideslip angle, and wind-axis roll rate will not be explicitly included, since it is assumed that these variables belong to the rigid body fast rotational dynamics and change over time scales sufficiently shorter than the time scales of the point mass translational dynamics so that the principles of time scale separation can be applied.

7.4 Optimal Recovery Guidance Law

An optimal guidance law was developed to perform the airspeed, flight path angle and bank angle recovery with the minimum of altitude loss while adhering to the constraints of the aircraft dynamics and structural limitations. Given the initial aircraft state, the guidance law generates the optimal reference trajectories for airspeed, flight path angle, and wind-axis bank angle, and the optimal sequence of angle of attack, wind-axis roll rate, and thrust commands to recover the aircraft to straight and level flight with an acceptable final airspeed while minimising the peak altitude lost during the recovery maneuver.

The guidance law is intended to be an outer-loop guidance law that provides the optimal state trajectories as reference signals for middle-loop feedback control laws that control the airspeed, the flight path angle and the bank angle, and provides the optimal control input sequences as feedforward reference signals for inner-loop feedback control laws that control the angle of attack, the wind-axis roll rate, and the thrust.

The outer-loop guidance law assumes that the inner-loop Lyapunov controller controls the rigid body fast rotational dynamics over time scales sufficienly shorter than the time scales of the point mass translational dynamics, so that the principles of time scale separation can be applied.

7.4.1 Requirements and Constraints

The following requirements and constraints are formulated for the optimal recovery guidance law:

- The aircraft must be recovered to straight and level flight (zero flight path angle, zero bank angle) with a minimum loss of altitude.
- During the recovery, the aircraft must not exceed the structural integrity envelope (maximum speed, maximum load factor).
- The aircraft must be recovered from unusual attitudes (high bank angles, high flight path angles), underspeed conditions, and overspeed conditions.
- The recovery must adhere to the constraints of the aircraft dynamics.
- The recovery must adhere to the constraints on admissible angle of attack commands, wind-axis roll rate commands, and thrust commands.
- The recovery must take the engine lag dynamics into account.
- As a secondary objective, the maximum speed reached during the recovery must be minimised.
- As a tertiary objective, the bank angle must be recovered to wings level as quickly as possible.

7.4.2 Overview and Architecture

The architecture of the proposed optimal recovery guidance law is shown in figure 7.4.

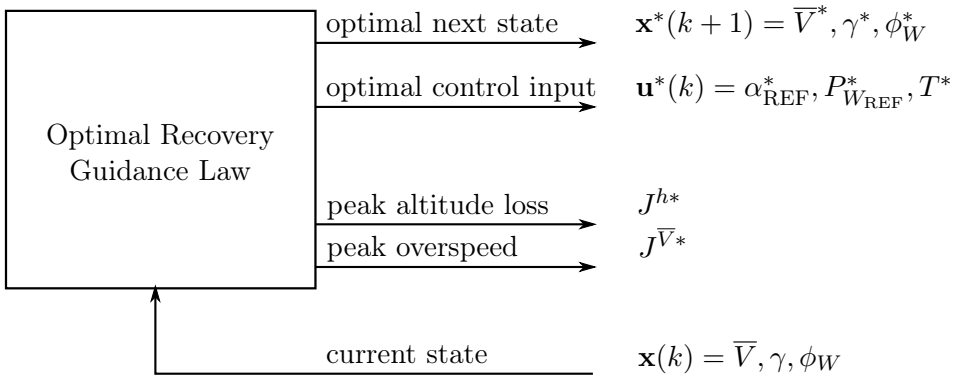


Figure 7.4: Optimal attitude and flight vector recovery architecture

The guidance law uses the current state $\mathbf{x}(k)$ as feedback and generates the optimal next state $\mathbf{x}^*(k)$ and the optimal control input $\mathbf{u}^*(k)$ to achieve the next state. It also reports the

peak altitude loss J^{h*} and the peak overspeed $J^{\overline{V}*}$ for optimal recovery from the current state. The aircraft translational state consists of the airspeed $\overline{V}$, flight path angle γ, and wind-axis bank angle ϕ_W and the control input consists of the angle of attack α_{ref}, the wind-axis roll rate command $P_{W_{\text{ref}}}$ and the thrust command T_C.

The guidance law obtains the optimal next state, the optimal control input, and the associated costs from a lookup table that was generated off-line using a dynamic programming algorithm.

7.4.3 Advantages of Dynamic Programming Solution

The dynamic programming approach has several advantages. Firstly, the dynamic programming algorithm searches a solution grid of the entire admissible state space, and effectively produces a lookup table of the optimal state trajectories and optimal control sequences for recovery from *all* possible initial states. Secondly, the lookup table explicitly stores the altitude that will be lost during optimal recovery for all states. Given the state of the aircraft, the lookup table therefore predicts how much altitude would be lost if the recovery procedure is activated immediately. Thirdly, although the dynamic programming search algorithm is computationally intensive, it produces a lookup table that is computationally light in comparison. This allows the computationally heavy "design" of the lookup table to be performed offline, and the precomputed lookup table to be used in a computationally light online implementation.

7.4.4 Dealing with Engine Dynamics

The engine dynamics typically do not operate over sufficiently shorter time scales relative to the point mass translational dynamics, and therefore cannot be omitted by exploiting time scale separation. The engine dynamics can be added to the translational dynamics by making the engine thrust a state instead of a control input, by using the engine thrust command as the control input, and by adding the first-order differential equation representing the engine lag dynamics to the equations of motion. However, this adds a fourth state variable to the state vector and makes the problem less tractable to be solved with dynamic programming.

Two approaches are therefore proposed to deal with the engine dynamics. If the engine lag dynamics is negligible, then it is simply omitted from the point mass translational dynamics. The engine thrust is then removed from the state vector and is treated as an input variable, as it is assumed that the engine thrust follows the engine thrust command instantaneously from the perspective of the point mass translational dynamics. Also, the first order differential equation representing the engine dynamics is removed from the set of differential equations. If the engine lag dynamics is significant, and cannot be omitted, then the engine thrust command is kept constant during the recovery process and is only set to trim the aircraft once the recovery has been completed. This allows the engine lag dynamics to be omitted, however at the cost of not being able to use the engine thrust as a variable control input. With both approaches, the engine thrust is omitted from the state vector, which reduces the dimension of the state vector to three state variables (airspeed, flight path angle, and wind-axis bank angle) and makes the problem tractable to be solved with dynamic programming.

To check the validity of the constant thrust approach, the optimal "longitudinal" recovery trajectories, where the bank angle is assumed to remain constant at wings level, were solved using both the constant thrust approach and an approach where the engine lag dynamics is included and the engine thrust is allowed to vary. Since the bank angle is assumed to remain constant, it can be omitted from the state vector and the wind axis roll rate command may be omitted from the input vector. This reduces the dimension of the state vector to three state variables (airspeed, flight path angle, and thrust) and makes the problem tractable to be solved with dynamic programming. The optimal recovery trajectories with constant thrust

were compared to the optimal recovery trajectories with variable thrust, but with engine lag dynamics, and it was found that the difference in the peak altitude lost was negligible.

7.4.5 Practical Controller Implementation

The dynamic programming algorithm is an optimal path planning algorithm, and not necessarily a robust feedback control law. For practical implementation, it is therefore recommended that the dynamic programming control law be supported with middle-loop feedback control laws that explicitly regulate airspeed, flight path angle and wind-axis bank angle.

The dynamic programming lookup table can then be used as a guidance law that generates reference trajectories for the airspeed, flight path angle, and wind-axis bank angle, and also generates feedforward control inputs for the angle of attack command, thrust command, and wind-axis roll rate command. The airspeed, flight path angle and wind-axis bank angle references are then provided to the middle-loop feedback control laws, and the feedforward commands for the angle of attack, thrust, and wind-axis roll rate are provided to the inner-loop control laws. The middle-loop feedback control laws will then provided additional disturbance rejection and robustness to model uncertainty, while the feedforward commands to the inner-loop controllers will compensate for the lag introduced by the middle-loop control laws.

If the dynamic programming outer-loop guidance law is used with the Lyapunov-based inner-loop control law, then middle-loop controllers should be added that control the flight path angle by commanding angle of attack, and control the wind-axis bank angle by commanding wind-axis roll rate. The optimal flight path angle and wind-axis bank angle trajectories would then be provided as references for the middle-loop control laws, and the optimal angle of attack and wind-axis roll rate commands would be provided as feedforward references for the inner-loop Lyapunov controller.

If the dynamic programming outer-loop guidance law is used with the conventional flight control laws, then the optimal flight path angle and wind-axis bank angle trajectories should be provided as reference trajectories for the existing middle-loop flight path angle and bank angle controllers, and the angle of attack and wind-axis roll rate commands should be converted to normal load factor and body-axis roll rate commands, and then provided as feedforward references for the inner-loop normal load factor and roll rate controllers.

7.5 The Optimal Control Problem

In this section, the attitude and flight vector recovery problem is formulated as an optimal control problem. In the following subsections, a brief background on optimal control theory is provided, and then the specific formulation of the attitude and flight vector recovery as an optimal control problem is presented. The primary source of the optimal control theory is the textbook on *Optimal Control Theory* by Kirk [62].

7.5.1 Optimal Control Background

Optimal control theory is concerned with finding a control law for a given system so that a specified optimality criterion is achieved. The optimality criterion is typically specified as a cost function to be minimised. The optimal control problem is therefore to find an optimal control law that produces the optimal control input signals and resulting optimal state trajectories that together minimise a given cost function, subject to a set of dynamic constraints, terminal state constraints, and state variable and control input inequality constraints. This can be expressed mathematically as follows

$$\mathbf{u}^*(t) = \arg\min_{\mathbf{u}(t)} J(\mathbf{x}(t), \mathbf{u}(t), t) \tag{7.18}$$

$$= \arg\min_{\mathbf{u}(t)} \left[h(\mathbf{x}(t_f)) + \int_{t_0}^{t_f} g(\mathbf{x}(t), \mathbf{u}(t), t) dt \right] \tag{7.19}$$

subject to the dynamic constraint

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), t) \tag{7.20}$$

the terminal state constraint

$$\mathbf{x}(t_f) \in \mathcal{X}_f \tag{7.21}$$

and the state space and input space constraints

$$\mathbf{x}(t) \in \mathcal{X} \tag{7.22}$$

$$\mathbf{u}(t) \in \mathcal{U} \tag{7.23}$$

In these equations, J is the cost function, $\mathbf{x}(t)$ is a candidate state trajectory, $\mathbf{u}(t)$ is a candidate control input signal, t is time, $\mathbf{f}()$ represents the set of nonlinear, time-variant differential equations of the system, $g()$ is the state transition cost function, and $h()$ is the terminal state cost function, $\mathcal{X}_\mathrm{f}$ is the set of admissable final states, and $\mathcal{X}$ and $\mathcal{U}$ are the sets of admissible states and admissible inputs. $\mathbf{x}^*(t)$ and $\mathbf{u}^*(t)$ are the optimal state trajectory and optimal control input signal that minimise the cost function subject to the constraints.

7.5.2 Formulation of Envelope Recovery as Optimal Control Problem

In this section, the attitude, flight path, and airspeed recovery problem is formulated mathematically as an optimal control problem. The dynamic system is defined as the reduced-order nonlinear differential equations describing the point mass translational dynamics of the aircraft, while the fast rotational dynamics is abstracted through time scale separation. The point mass translational dynamics model therefore includes the wide-envelope aerodynamic models describing the aerodynamic forces, but not the aerodynamic moments. The airspeed, flight path angle, and bank angle limits are specified by defining the sets of admissible states. The control input limits are specified by defining sets of admissible inputs. The requirement to recover the aircraft to straight and level flight (zero flight path angle, zero bank angle) and to within an acceptable airspeed range is translated into a set of admissible final states. The objectives to minimise the total altitude lost, to minimise the maximum airspeed reached, and to return the bank angle to zero as quickly as possible are translated into a novel hierarchical multi-objective cost function. The hierarchical cost function allows the primary objective to be prioritised without making any compromises to the secondary or tertiary objectives. (In contrast, a single multi-objective cost function would typically lead to trade-offs between the primary, secondary, and tertiary objectives to minimise the total cost based on some weighting scheme.) The problems of underspeed and overspeed recovery are included in the problem formulation by including underspeed and overspeed states in the range of admissible states, but only including acceptable speeds in the set of admissible *final* states.

The problem formulation also incorporates the requirement to adhere to the structural integrity envelope in terms of airspeed and normal load factor. The airspeed constraint is

incorporated as a state constraint on airspeed. The load factor constraint is incorporated in a novel way by translating it into a state-dependent input constraint, where the set of admissible angle of attack control inputs is limited as a function of the current airspeed state.

Finally, the problem formulation can also take the limitations of the inner-loop Lyapunov controller into account. For example, we may wish to constrain the maximum admissible wind-axis roll rate command at high angles of attack. This can be accomplished with a coupled control input constraint, where the set of admissible wind-axis roll rate commands is limited as a function of the angle of attack command.

The optimal attitude, flight path, and airspeed recovery problem is now formulated mathematically as follows

Point Mass Translational Dynamics

The dynamic system is defined as the reduced-order nonlinear differential equations describing the point mass translational dynamics of the aircraft

$$\dot{\overline{V}} = \frac{1}{m}[T - \frac{1}{2}\rho\overline{V}^2 SC_D(\alpha) - mg\sin(\gamma)] \tag{7.24}$$

$$\dot{\gamma} = \frac{1}{\overline{V}}[\frac{1}{2}\rho\overline{V}^2 SC_L(\alpha)\cos\Phi_W - mg\cos(\gamma)]\frac{1}{m} \tag{7.25}$$

$$\dot{\Phi}_W = P_W \tag{7.26}$$

These reduced-order equations have been obtained by making the following simplifications:

1. The flight path heading ψ_W is left out of the state vector, because it does not couple back into the other states, and does not contribute to the cost function to be minimised. The objective is to recover the airspeed, flight path angle and bank angle of the aircraft with minimum altitude loss, and it does not matter what happens to the flight path heading during the recovery process.

2. The sideslip angle β will not be used as a control input, and the inner-loop Lyapunov controller will be commanded to regulate the sideslip angle to zero during the upset recovery. This means that the aerodynamic side force Y and the dependencies of the aerodynamic lift coefficient C_L and the aerodynamic drag coefficient C_D on the sideslip angle β may be omitted. (However, it might be an interesting topic of future research to consider how the sideslip angle may be used to produce additional drag to control the velocity magnitude, and may be used to produce side force to control the flight path angle when at high bank angles.)

State Vector

The state vector $\mathbf{x}$ is defined as

$$\mathbf{x} = \begin{bmatrix}\overline{V} & \gamma & \phi_W\end{bmatrix}^T \tag{7.27}$$

where the state variables are airspeed $\overline{V}$, flight path angle γ, and wind-axis bank angle ϕ_W. The flight path heading ψ_W has been left out of the state vector, because it does not couple back into any of the other states, and will not play a role in the recovery cost function.

Control Input Vector

The control input vector $\mathbf{u}$ is defined as

$$\mathbf{u} = \begin{bmatrix} \alpha & P_W & T \end{bmatrix}^T \tag{7.28}$$

where the control input variables are selected as angle of attack α, wind-axis roll rate P_W, and thrust T. The sideslip angle β has been left out of the control input vector, because it will not be used as a control input and is assumed to be regulated to zero by the inner-loop Lyapunov controller.

State Constraints

The physical limitations on airspeed, flight path angle, and bank angle are specified by defining the following sets of admissible states.

$$\overline{V} \in [\overline{V}_{\min}, \overline{V}_{\max}] \tag{7.29}$$
$$\gamma \in [\gamma_{\min}, \gamma_{\max}] \tag{7.30}$$
$$\phi_W \in [-\phi_{W\max}, \phi_{W\max}] \tag{7.31}$$

The range of *admissible* airspeed values extends beyond the range of *acceptable* airspeeds, and include airspeed values that represent both underspeed and overspeed conditions. The minimum admissible airspeed $\overline{V}_{\min}$ is selected as the lowest airspeed from which the aircraft may be expected to recover, and may even be below the stall speed. The maximum admissible airspeed $\overline{V}_{\max}$ is defined by the upper airspeed bound of the structural integrity envelope.

Input Constraints

The control input limits are specified by defining the following sets of admissible inputs.

$$\alpha \in [\alpha_{\min}, \alpha_{\max}] \tag{7.32}$$
$$T \in [T_{\min}, T_{\max}] \tag{7.33}$$
$$P_W \in [-P_{W\max}, P_{W\max}] \tag{7.34}$$

The load factor constraint is incorporated in a novel way by translating it into a state-dependent input constraint, where the set of admissible angle of attack control inputs is a function of the current airspeed state.

$$n_{L\min} \leq \frac{\frac{1}{2}\rho\overline{V}^2 S C_L(\alpha)}{m} \leq n_{L\max} \tag{7.35}$$

where $n_{L\min}$ and $n_{L\max}$ are the minimum and maximum normal load factor allowed by the structural integrity envelope.

We may also constrain the maximum admissible wind-axis roll rate command at high angles of attack using the following coupled control input constraint, where the set of admissible wind-axis roll rate commands is limited as a function of the angle of attack command.

$$P_{W\min}(\alpha) \leq P_W \leq P_{W\max}(\alpha) \tag{7.36}$$

Terminal State Constraints

The requirement to recover the aircraft to straight and level flight (zero flight path angle, zero bank angle) and to within an acceptable airspeed range is translated into the following set of admissible final states.

$$\overline{V}(t_f) \in [\overline{V}_{f\min}, \overline{V}_{f\max}] \tag{7.37}$$
$$\gamma(t_f) = 0 \tag{7.38}$$
$$\phi_W(t_f) = 0 \tag{7.39}$$

where $\overline{V}_{f\min}$ and $\overline{V}_{f\max}$ are the minimum and maximum acceptable *final* airspeeds, respectively.

Hierarchical Multi-Objective Cost Function

The primary objective to minimise the total altitude lost, the secondary objective to minimise the maximum airspeed reached, and the tertiary objective to return the bank angle to zero as quickly as possible are translated into the following novel hierarchical multi-objective cost function. The primary cost function is the total altitude lost during the recovery, defined as

$$J^h = \int_{t_0}^{t_f} \max(-\dot{h}(t), 0)dt \tag{7.40}$$
$$= \int_{t_0}^{t_f} \max(-\overline{V}(t)\sin\gamma(t), 0)dt \tag{7.41}$$

It is important to note that the cost function has been defined so that a negative climb rate increases the cost, but that a positive climb rate does not reduce the cost. If we allow the cost to be reduced by gaining altitude, then the optimisation algorithm is free to lose any amount of altitude, as long as it "makes up for it" by gaining an equal amount of altitude during a different stage of the recovery, so that the total integral of the climb rate over time is minimised. By only allowing negative climb rates to contribute to the altitude cost, the cost function represents the maximum altitude lost during the recovery. The secondary cost function is the maximum velocity reached over the course of the recovery, defined as

$$J^{\overline{V}} = \max(\overline{V}(t)),\ t \in [t_0, t_f] \tag{7.42}$$

The tertiary cost function is the time integral of the absolute value of the bank angle, defined as

$$J^{\phi} = \int_{t_0}^{t_f} \|\phi_W(t)\|\, dt \tag{7.43}$$

The hierarchical cost function is minimised as follows: First, the primary cost function J^h is minimised without considering the secondary or tertiary cost functions. Next, if more than one solution minimises the primary cost function, then the solutions that also minimise the secondary cost function $J^{\overline{V}}$ are selected from among the solutions that already minimise the primary cost function. Finally, if more than one solution minimise both the primary and secondary cost functions, then a solution that also minimises the tertiary cost function J^{ϕ} is selected from among the solutions that already minimise both the primary and secondary cost functions.

The hierarchical cost function allows the primary objective to be prioritised without making any compromises to the secondary or tertiary objectives. (In contrast, a single multi-objective cost function would typically lead to trade-offs between the primary, secondary, and tertiary objectives to minimise the total cost based on some weighting scheme.)

The motivation is that minimising the total altitude lost is of utmost importance, and that we do not want to trade any altitude loss to reduce the maximum airspeed reached or to reduce the time it takes to return the bank angle to wings level. However, if more than one solution would minimise the total altitude lost, then we would prefer the solution that minimises the maximum airspeed reached during the recovery. Furthermore, if more than one solution minimises both the altitude lost and the maximum airspeed reached, then we would prefer the solution that also returns that aircraft to wings level in the least amount of time.

Now that the optimal control problem has been formulated mathematically, we can proceed with solving the problem to obtain the optimal sequence of control inputs that would result in the optimal state trajectories.

7.6 The Dynamic Programming Solution

The two main approaches to solving optimal control problems are *dynamic programming* and the *calculus of variations* approach. Dynamic programming is a search algorithm that models an optimal control problem as a multi-stage decision process and uses the principle of optimality to find the optimal state trajectories and control inputs to minimise the cost function from all initial states. Dynamic programming uses the principle of optimality to drastically reduce the number of calculations required to determine the optimal control law. Unfortunately, for high-dimensional systems the memory requirements of the dynamic programming algorithm becomes prohibitive. This is called "the curse of dimensionality".

The calculus of variations approach transforms the optimal control problem into a nonlinear two-point boundary-value problem that cannot be solved analytically, and is challenging to solve numerically due to the fact that the boundary conditions are split. However, the optimal controls and trajectories can usually be solved using iterative numerical techniques, such as the method of steepest descent, variation of extremals, quasilinearisation, and gradient projection.

It was decided to use dynamic programming to solve the optimal recovery problem. The main reason is that the dynamic programming approach produces a lookup table of the optimal state trajectories and optimal control sequences for recovery from *all* possible initial states. In contrast, the calculus of variations approach produces a single optimal trajectory and control sequence for a given initial state, and must therefore be repeated for each different initial state. With dynamic programming, the computationally heavy search algorithm that produces the lookup table is executed offline, and the pre-computed lookup table can then be used in a computationally light online implementation. With the calculus of variations approach, the computationally heavy iterative numerical solution to obtain the optimal trajectory and control sequence must be performed online given the initial state. Finally, the lookup table produced by the dynamic programming approach explicitly stores the altitude that will be lost during optimal recovery for all states. Given the state of the aircraft, the lookup table therefore predicts how much altitude would be lost if the recovery procedure is activated immediately.

In the following subsections, an overview of the general dynamic programming approach will be provided, and the specific application of the dynamic programming approach to the optimal recovery problem will be presented.

7.6.1 Dynamic Programming Background

This section provides an overview of the general dynamic programming approach to solving optimal control problems. The primary source for the dynamic programming theory is the

textbook on *Optimal Control Theory* by Kirk [62].

Dynamic programming is a numerical algorithm that models an optimal control problem as a multi-stage decision process and uses the principle of optimality to solve for the optimal state trajectories and control inputs. The principle of optimality states that if the optimal state trajectory from a given state to a terminal state passes through an intermediate state, then the optimal trajectory from the intermediate state to the terminal state is a subsection of the optimal trajectory from the first state.

To apply dynamic programming to an optimal control problem, the problem has to be discretised in time to represent different decision stages, and must be discretised in state to represent a finite number of decisions to be made at each decision stage. The dynamic programming algorithm starts at the terminal stage and works backward in time through intermediate stages until it finds the optimal admissible path from the initial state to a terminal state. The algorithm starts at the final stage and assigns a termination cost to each admissible terminal state. The algorithm then steps back one discrete time step to the previous stage, iterates through all the discrete states, and for each state determines the admissible control input that will transition the system from this state to one of the terminal states with the minimum total cost. The total cost to the end is the sum of the cost to transition from the current state to the terminal state and the cost of the particular terminal state. For each state that has a path to the end, the optimal path cost from that state to a terminal state is stored.

If, after iterating through all the states in this time instant, there are still states that do not have an admissible path to a terminal state, then the algorithm continues. The algorithm steps back another stage, iterates through all the discrete states, and for each state determines the admissible control input that will transition the system to an admissible state in the next intermediate stage while minimising the total path cost from this state to a terminal state. The cost from the current state to the end is now the sum of the optimal cost from the intermediate state to the end and the incremental cost to transition from the current state to the intermediate state. If, after iterating through all the states in this stage, there are still states that do not have a path to a terminal state, even through intermediate states, then the algorithm steps back another stage. The process of going back discrete time states is repeated until optimal paths have been found from all discrete states to a terminal state, or until no new admissible paths can be found, or until a maximum number of time steps have been reached.

Optimal Control Problem

We wish to find the optimal state trajectory $\mathbf{x}^*(t)$ and the optimal control input signal $\mathbf{u}^*(t)$ from a given initial state $\mathbf{x}(t_0) = \mathbf{x}_{\text{initial}}$ to minimise the following cost function

$$J = \left[h(\mathbf{x}(t_f)) + \int_{t_0}^{t_f} g(\mathbf{x}(t), \mathbf{u}(t), t)dt\right] \tag{7.44}$$

subject to the dynamic constraint

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), t) \tag{7.45}$$

the terminal state constraint

$$\mathbf{x}(t_f) \in \mathcal{X}_f \tag{7.46}$$

and the state space and input space constraints

$$\mathbf{x}(t) \in \mathcal{X} \tag{7.47}$$
$$\mathbf{u}(t) \in \mathcal{U} \tag{7.48}$$

In these equations, J is the cost function, $h()$ is the terminal state cost function, $g()$ is the state transition cost function, $\mathbf{x}(t)$ is a candidate state trajectory, $\mathbf{u}(t)$ is a candidate control input signal, t is time, $\mathbf{f}()$ represents the set of nonlinear, time-variant differential equations of the system, $\mathcal{X}$ and $\mathcal{U}$ are the sets of admissible states and admissible inputs, and $\mathcal{X}_\mathrm{f}$ is the set of admissable final states.

Quantised State Vector Array

The continuous set of admissible states $\mathcal{X}$ is discretised to create a finite set of state values for the dynamic programming algorithm. We therefore create an array $\mathbf{X}_q$ of quantised state vector values $\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_n$ that span the set of admissible states $\mathcal{X}$.

$$\mathbf{X}_q = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_i, \ldots, \mathbf{x}_n\} \tag{7.49}$$

Discrete-Time Dynamic Model

The continuous-time differential equations describing the system dynamics are discretised to produce discrete-time difference equations that describe the state transition from one discrete time instant to the next.

$$\mathbf{x}(k+1) = \mathbf{x}(k) + \mathbf{f}(\mathbf{x}(k), \mathbf{u}(k), k)\Delta t \tag{7.50}$$

where Δt is the sampling period of the discrete time step.

Control Inputs

The continuous set of admissible inputs $\mathcal{U}$ must be discretised to create a finite set of input values. We therefore create an input array $\mathbf{U}_q$ of quantised input vector values $\mathbf{u}_0, \mathbf{u}_1, \ldots, \mathbf{u}_n$ that span the set of admissible inputs $\mathcal{U}$.

$$\mathbf{U}_q = \{\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_i, \ldots, \mathbf{u}_m\} \tag{7.51}$$

Terminal State Cost

The terminal state cost of each state in the set of admissible final states is assigned using the termination cost function $h()$

$$J^*(\mathbf{x}_j) = h(\mathbf{x}_j) \tag{7.52}$$

Incremental State Transition Cost Function

The continuous-time cost function is discretised by expressing the total path cost J_{ij} from the current state $\mathbf{x}_i(k)$ via the next state $\mathbf{x}_j(k+1)$, as the sum of the incremental cost ΔJ_{ij} of transitioning from the current state to the next state, and the total path cost of the next state J_j to a final state.

$$J_{ij}(\mathbf{x}_i, \mathbf{x}_j) = \Delta J_{ij}(\mathbf{x}_i, \mathbf{x}_j) + J_j(\mathbf{x}_j) \tag{7.53}$$

The incremental state transition cost ΔJ_{ij} is obtained by discretising the state transition cost function $g(\mathbf{x}(t), \mathbf{u}(t), t)$ as follows

$$\Delta J_{ij}(\mathbf{x}_i(k), \mathbf{x}_j(k+1)) \approx g(\mathbf{x}_i(k), \mathbf{u}_{ij}(k), k)\Delta t \tag{7.54}$$

where $\mathbf{u}_{ij}$ is the admissible control input that transitions the system from state $\mathbf{x}_i$ at time k to state $\mathbf{x}_j$ at time $k+1$.

Algorithm Execution

First, we create a grid of discrete time instants and discrete state values on which the optimal control problem will be solved. The grid is represented by data structures for the optimal cost $J^*_{i,k}$, the optimal next state index $j^*_{i,k}$, and the optimal control input $\mathbf{u}^*_{i,k}$ from each state $\mathbf{x}_i$ at time instant k.

$$\mathbf{J}^*_{n\times N} = \{J^*_{i,k}\} \tag{7.55}$$
$$\mathbf{j}^*_{n\times N} = \{j^*_{i,k}\} \tag{7.56}$$
$$\mathbf{U}^*_{n\times N} = \{\mathbf{u}^*_{i,k}\} \tag{7.57}$$

with $i \in [1, n]$ and $k \in [1, N]$, where n is the number of quantised state values, and N is the number of discrete time steps.

We start at the final time step $k = N$, and assign a termination cost for each admissible final state

$$J^*_{i,N} \leftarrow h(\mathbf{x}_i), \text{ if } \mathbf{x}_i \in \mathcal{X}_\mathrm{f} \tag{7.58}$$

Now we go back one time instant, and consider all possible transitions from state values at this time instant $k = N-1$ to all admissible final state values at the final time instant $k = N$. For each state, we check if an admissible control input exists that can transition the system from this state value to an admissible final state value within one discrete time step. If an admissible control exists, then we calculate the incremental transition cost of moving from this state to the particular final state. Given the state transition equation

$$\mathbf{x}_j(k+1) = \mathbf{x}_i(k) + \mathbf{f}(\mathbf{x}(k), \mathbf{u}_{ij}(k), k)\Delta t \tag{7.59}$$

we calculate the control input $\mathbf{u}_{ij}(k)$ that would transition the system from the current state $\mathbf{x}_i(k)$ to the next state $\mathbf{x}_j(k+1)$. If $\mathbf{u}_{ij}(k)$ is an admissible control input, that is if $\mathbf{u}_{ij}(k) \in \mathcal{U}$, then we calculate the total path cost $J_{i,N-1}$ from the current state $\mathbf{x}_i$ to the final state $\mathbf{x}_j(N)$ as the sum of the incremental state transition cost ΔJ_{ij} and the terminal cost $h(\mathbf{x}_j(N))$ of the final state.

$$J_{ij,N-1}(\mathbf{x}_i, \mathbf{x}_j(N)) = \Delta J_{ij}(\mathbf{x}_i, \mathbf{x}_j(N)) + h(\mathbf{x}_j(N)) \tag{7.60}$$

If the total path cost $J_{ij,N-1}$ is lower than the lowest path cost $J^*_{i,N-1}$ so far, then it becomes the new lowest path cost

$$J^*_{i,N-1} \leftarrow J_{ij,N-1} \tag{7.61}$$

Also, the control input $\mathbf{u}_{ij}(k)$ and the next state index j become the new best control input $\mathbf{u}^*_{i,k}$ and the new best next state $j^*_{i,k}$ for the current state $\mathbf{x}_i(k)$.

$$\mathbf{u}^*_{i,k} \leftarrow \mathbf{u}_i(k)$$
$$j^*_{i,k} \leftarrow j$$

We repeat this process until we have iterated through all possible transitions from state values at this time instant to all admissible final state values at the final time instant. If there are still states that do not have admissible paths to a final state, we go back another time instant. We then consider all possible transitions from state values at time instant k to all intermediate state values at the next time instant $k+1$.

For each current state $\mathbf{x}_i(k)$, we check if an admissible control input exists that can transition the system to the next state $\mathbf{x}_j(k)$ within one discrete time step. If an admissible control input exists, then we calculate the total path cost $J_{ij,k}$ from the current state to a final state via the next state. The total path cost is calculated as the incremental state transition cost ΔJ_{ij} from the current state to the next state plus the lowest total path cost $J^*_{j,k+1}(\mathbf{x}_j)$ from the next state to a final state.

$$J_{ij,k}(\mathbf{x}_i, \mathbf{x}_j) = \Delta J_{ij}(\mathbf{x}_i, \mathbf{x}_j) + J^*_{j,k+1}(\mathbf{x}_j) \tag{7.62}$$

If the total path cost $J_{ij,k}$ via the next state is lower than the lowest path cost $J^*_{i,k}$ from the current state so far, then it becomes the new lowest path cost from the current state

$$J^*_{i,k} \leftarrow J_{ij,k} \tag{7.63}$$

and the control input $\mathbf{u}_{ij}(k)$ and the next state index j become the new best control input $\mathbf{u}^*_{i,k}$ and the new best next state $j^*_{i,k}$ for the current state.

$$\begin{aligned} \mathbf{u}^*_{i,k} &\leftarrow \mathbf{u}_i(k) \\ j^*_{i,k} &\leftarrow j \end{aligned}$$

We repeat this process until we have iterated through all possible transitions from state values at the current time instant to all state values at the next time instant. If there are still states that do not have admissible paths to a final state, then we go back another time instant.

The process of going back a time step and iterating through all possible transitions is repeated until admissible paths have been found from all states, or until no new admissible paths with lower path costs are found for an entire time step iteration, or until a maximum number of time step iterations is reached.

Optimal State Trajectories and Control Input Sequences

The data structures that store the optimal cost grid $\mathbf{J}^*_{n\times N}$, the optimal next state index grid $\mathbf{j}^*_{n\times N}$, and the optimal control input grid $\mathbf{U}^*_{n\times N}$ now represent lookup tables that can be used to generate the optimal state trajectories and optimal control sequences from all admissible initial states. For a given state $\mathbf{x}_i(k)$, the optimal next state index is $j^*_{i,k}$, the optimal next state is $\mathbf{x}^*_j(k+1) = \mathbf{X}_q(j^*_{i,k})$, and the optimal control input is $\mathbf{u}(k) = \mathbf{u}^*_{i,k}$. The optimal cost from the given state is $J^*_{i,k}$.

7.6.2 Design Decisions

The following design decisions have been made:

1. The engine thrust shall be kept constant during the recovery process and shall only be used to trim the aircraft once the recovery has been completed. The engine is a very low bandwidth actuator and will not be able to contribute effectively to the relatively high actuation bandwidth required by the recovery dynamics. This design decision also effectively allows us to ignore the engine lag dynamics. This means that the engine state does not have to be included in the state vector, which reduces the dimension of the state vector by one, making the problem more tractable for the dynamic programming search algorithm.

2. The point mass translational dynamics will be discretised in time using a sampling period of one second. The inner-loop Lyapunov controller has a time constant of about 0.2 seconds which is sufficiently shorter than one second so that the outer-loop controller can assume time scale separation. (This assumption will be validated in simulation.)

7.6.3 Dynamic Programming Implementation

In this subsection, the dynamic programming approach is applied to the optimal recovery problem.

Quantised State Vector Array

We start by creating a grid of quantised state vector values $\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_n$ and arranging them into an array $\mathbf{X}_\mathrm{q}$ that contains every combination of the quantised airspeed, flight path angle, and wind-axis bank angle values.

$$\begin{aligned} \mathbf{X}_\mathrm{q} &= \{\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_n\} && (7.64)\\ &= \overline{\mathbf{V}}_q \times \mathbf{\Gamma}_q \times \mathbf{\Phi}_{\mathbf{W}q} && (7.65)\\ &= \{(\overline{V}, \gamma, \phi_W) : \overline{V} \in \overline{\mathbf{V}}_q, \gamma \in \mathbf{\Gamma}_q, \phi_\mathrm{W} \in \mathbf{\Phi}_{\mathbf{W}q}\} && (7.66) \end{aligned}$$

The quantised state vector array $\mathbf{X}_\mathrm{q}$ is the Cartesian product of the quantised airspeed array $\overline{\mathbf{V}}_q$, the quantised flight path angle array $\mathbf{\Gamma}_q$, and the quantised bank angle array $\mathbf{\Phi}_{\mathbf{W}q}$ defined by the following sets

$$\begin{aligned} \overline{\mathbf{V}}_q &= \{\overline{V}_\mathrm{min} : \Delta\overline{V} : \overline{V}_\mathrm{max}\} && (7.67)\\ \mathbf{\Gamma}_q &= \{\gamma_\mathrm{min} : \Delta\gamma : \gamma_\mathrm{max}\} && (7.68)\\ \mathbf{\Phi}_{\mathbf{W}q} &= \{\phi_{W\,\mathrm{min}} : \Delta\phi_W : \phi_{W\,\mathrm{max}}\} && (7.69) \end{aligned}$$

where $\Delta\overline{V}$, $\Delta\gamma$, and $\Delta\phi_W$ are the quantisation intervals for the airspeed, flight path angle and bank angle respectively. The set notation has been borrowed from the colon operator used in Matlab to create regularly spaced vectors. The length of the state vector array is the product of the respective lengths of the airspeed array, the flight path angle array and the bank angle array. An element in the state vector array is addressed by its index in the array, as follows

$$\mathbf{x}_i = [\overline{V}_i, \gamma_i, \phi_{W\,i}]^T \tag{7.70}$$

where $i \in [1, n]$ is the array index, and n is the length of the state vector array. By iterating through the index i, every possible combination of airspeed, flight path angle and bank angle in the quantised set of admissible states can be addressed.

Discrete-Time Dynamic Model

The differential equations describing the point mass translational dynamics are discretised to produce the following discrete-time difference equations that describe the state transition from one discrete time instant to the next.

$$\begin{aligned} \overline{V}(k+1) &= \overline{V}(k) + \frac{1}{m}[T(k) - \frac{1}{2}\rho\overline{V}^2(k)SC_D(\alpha(k)) - mg\sin\gamma(k)]\Delta t && (7.71)\\ \gamma(k+1) &= \gamma(k) + \frac{1}{m\overline{V}(k)}[\frac{1}{2}\rho\overline{V}^2(k)SC_L(\alpha(k))\cos\phi_W(k) - mg\cos\gamma(k)]\Delta t && (7.72)\\ \phi_W(k+1) &= \phi_W(k) + P_W(k)\Delta t && (7.73) \end{aligned}$$

If the engine dynamics is included, then the following difference equation must also be added

$$T(k+1) = e^{-\Delta t/\tau}T(k) + (1 - e^{-\Delta t/\tau})T_c(k) \tag{7.74}$$

Control Inputs

It was decided not to quantise the control input vector $\mathbf{u}$. Since the dynamic system has three states and three inputs, we can quantise the state vector, and then simultaneously solve the set of difference equations to calculate the exact control input $\mathbf{u}$ required to transition the system between two given states in one time step. We can then check whether the calculated control input is admissible, and by association check whether the given state transition is admissible. This is a key realisation, because it makes the problem more tractable to be solved by dynamic programming. If we had been forced to quantise both the state vector and the control input vector, then the search space becomes much larger.

Given two quantised states $\mathbf{x}_i$ at time instant k and $\mathbf{x}_j$ at the next time instant $k+1$, the control input required to transition from $\mathbf{x}_i$ to $\mathbf{x}_j$ in one time step Δt may be calculated as follows

$$\alpha_i(k) = C_L^{-1}\left(\frac{m\overline{V}_i(k)\frac{\gamma_j(k+1)-\gamma_i(k)}{\Delta t} + mg\cos\gamma_i(k)}{\frac{1}{2}\rho\overline{V}_i^2(k)S\cos\phi_{Wi}(k)}\right) \tag{7.75}$$

$$T_i(k) = \frac{m[\overline{V}_j(k+1) - \overline{V}_i(k)]}{\Delta t} + \frac{1}{2}\rho\overline{V}_i^2(k)SC_D(\alpha_i(k)) + mg\sin\gamma_i(k) \tag{7.76}$$

$$P_{Wi}(k) = \frac{\phi_{Wj}(k+1) - \phi_{Wi}(k)}{\Delta t} \tag{7.77}$$

In other words, we calculate the angle of attack command that will transition the current flight path angle to the next flight path angle, the thrust command that will transition the current airspeed to the next airspeed, and the wind-axis roll rate command that will transition the current wind-axis bank angle to the next wind-axis bank angle.

Notice that the first equation contains a singularity when either the airspeed is zero or the bank angle is at 90 degrees. This represents the fact that the aircraft cannot produce lift at zero airspeed and the fact that the lift vector cannot affect the flight path angle when its direction is perpendicular to the vertical. The singularity at zero speed is avoided by not including zero airspeed in the set of admissible states. However, we would like to consider a bank angle of 90 degrees as an admissible state, since we may want to recover the aircraft from an inverted attitude. We will therefore use the following alternate set of equations for solving the required control input when the bank angle is at $\pm$ 90 degrees.

$$T_i(k) = T_{\text{trim}} \tag{7.78}$$

$$\alpha_i(k) = C_D^{-1}\left(\frac{T_i(k) - \frac{m[\overline{V}_j(k+1)-\overline{V}_i(k)]}{\Delta t} - mg\sin\gamma_i(k)}{\frac{1}{2}\rho\overline{V}_i^2(k)S}\right) \tag{7.79}$$

$$P_{Wi}(k) = \frac{\phi_{Wj}(k+1) - \phi_{Wi}(k)}{\Delta t} \tag{7.80}$$

Basically, we calculate the angle of attack command and the thrust command, by setting the thrust command to the trim thrust and then calculating the angle of attack that would produce sufficient drag to transition the current airspeed to the next airspeed. (The wind-axis roll rate is calculated to transition the current wind-axis bank angle to the next wind-axis bank

angle, as before.) Since the flight path angle is essentially uncontrolled, we use the difference equation to predict what the flight path angle at the next time step will be and check if it is close enough to the flight path angle of the state to which we are transitioning.

$$\gamma_j(k+1) = \gamma_i(k) + \frac{[\frac{1}{2}\rho\overline{V}_i^2(k)SC_L(\alpha_i(k))\cos\phi_{Wi}(k) - mg\cos\gamma_i(k)]\Delta t}{m\overline{V}_i(k)} \tag{7.81}$$

If the predicted flight path angle is close enough to the flight path angle of the next state, and the angle of attack command, thrust command, and wind-axis roll rate command are all admissible inputs, then the state transition is considered admissible.

Terminal State Constraints

The set of admissible final states is also quantised as follows

$$\mathbf{X}_{f_q} = \{(\overline{V}, \gamma, \phi_W) : \overline{V} \in \{\overline{V}_{f\min} : \Delta\overline{V} : \overline{V}_{f\max}\}, \gamma = 0, \phi_W = 0\} \tag{7.82}$$

The quantised set of admissible final states $\mathbf{X}_{f_q}$ is a subset of the quantised set of all admissible states $\mathbf{X}_q$ contained in the state vector array. It would therefore be convenient if we could label all the elements of the state vector array as being either "a final state" or "not a final state". We will accomplish this by assigning a termination cost of zero to all states that are admissible final states, and a termination cost of infinity to all states that are not admissible final states, as follows

$$h(\mathbf{x}_i) = \left\{ \begin{array}{ll} 0, & \overline{V}_i \in [\overline{V}_{f\min}, \overline{V}_{f\max}] \text{ and } \gamma_i = 0 \text{ and } \phi_{Wi} = 0 \\ \infty, & \text{otherwise} \end{array} \right\} \tag{7.83}$$

where $h()$ is the termination cost function.

Incremental State Transition Cost Functions

The hierarchical multi-objective cost function that consists of the primary cost function in terms of altitude loss, the secondary cost function in terms of maximum airspeed, and the tertiary cost function in terms of the time integral of the bank angle, are all discretised and translated into incremental state transition cost functions.

The incremental transition cost of the primary cost function ΔJ^h is obtained as

$$\Delta J^h(\mathbf{x}_i(k), \mathbf{x}_j(k+1)) = \max(-\overline{V}_{\text{avg}}\sin\gamma_{\text{avg}}, 0)\Delta t \tag{7.84}$$

with

$$\overline{V}_{\text{avg}} = \frac{\overline{V}_i(k) + \overline{V}_j(k+1)}{2} \tag{7.85}$$

$$\gamma_{\text{avg}} = \frac{\gamma_i(k) + \gamma_j(k+1)}{2} \tag{7.86}$$

The total secondary cost $J^{\overline{V}}$ of transitioning from state $\mathbf{x}_i$ at time instant k to state $\mathbf{x}_j$ at time instant $k+1$ is given by

$$J^{\overline{V}}(\mathbf{x}_i(k), \mathbf{x}_j(k+1)) = \max(\|\overline{V}_i\|, J^{\overline{V}*}(\mathbf{x}_j(k+1))) \tag{7.87}$$

which simply means that the total secondary cost is the greater of the airspeed of state $\mathbf{x}_i$ and the maximum airspeed of the optimal trajectory from state $\mathbf{x}_j$ to a final state.

The incremental transition cost of the tertiary cost function J^{ϕ} is obtained as

$$\Delta J^{\phi}(\mathbf{x}_i(k), \mathbf{x}_j(k+1)) = \|\phi_{W\,\text{avg}}\|\Delta t \tag{7.88}$$

with

$$\phi_{W\,\text{avg}} = \frac{\phi_{W\,i}(k) + \phi_{W\,j}(k+1)}{2} \tag{7.89}$$

Initialisation

1. Create data structures for the quantised state vector array $\mathbf{X}_q$, the optimal altitude cost array $\mathbf{J}^{h*}$, the optimal airspeed cost array $\mathbf{J}^{\overline{V}*}$, the optimal bank angle cost array $\mathbf{J}^{\phi*}$, the optimal next state index array $\mathbf{j}^*$, and the optimal control input array $\mathbf{U}^*$

$$\begin{aligned}
\mathbf{X}_q &= \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\} && (7.90)\\
\mathbf{J}^{h*} &= \{J_1^{h*}, J_2^{h*}, \ldots, J_n^{h*}\} && (7.91)\\
\mathbf{J}^{\overline{V}*} &= \{J_1^{\overline{V}*}, J_2^{\overline{V}*}, \ldots, J_n^{\overline{V}*}\} && (7.92)\\
\mathbf{J}^{\phi*} &= \{J_1^{\phi*}, J_2^{\phi*}, \ldots, J_n^{\phi*}\} && (7.93)\\
\mathbf{j}^* &= \{j_1^*, j_2^*, \ldots, j_n^*\} && (7.94)\\
\mathbf{U}^* &= \{\mathbf{u}_1^*, \mathbf{u}_2^*, \ldots, \mathbf{u}_n^*\} && (7.95)
\end{aligned}$$

2. Populate the state vector array $\mathbf{X}_q$ with the Cartesian product of the quantised airspeed array $\overline{\mathbf{V}}_q$, the quantised flight path angle array $\mathbf{\Gamma}_q$, and the quantised bank angle array $\mathbf{\Phi}_{\mathbf{W}q}$, as follows

$i \leftarrow 1$
for $l = 1$ to length($\overline{\mathbf{V}}_q$) **do**
  $\overline{V}_i \leftarrow \overline{\mathbf{V}}_q(l)$
  for $m = 1$ to length($\mathbf{\Gamma}_q$) **do**
    $\gamma_i \leftarrow \mathbf{\Gamma}_q(m)$
    for $n = 1$ to length($\mathbf{\Phi}_{\mathbf{W}q}$) **do**
      $\phi_{W\,i} \leftarrow \mathbf{\Phi}_{\mathbf{W}q}(n)$
      $\mathbf{x}_i \leftarrow \begin{bmatrix}\overline{V}_i & \gamma_i & \phi_{W\,i}\end{bmatrix}^T$
      $i \leftarrow i + 1$
    end for
  end for
end for

3. Start at the final time instant

$k \leftarrow N$

4. Iterate through all the elements of the state vector array from the first state index to the last state index, and do the following:

5. For each state $\mathbf{x}_i$, populate the corresponding optimal altitude cost J_i^{h*}, the corresponding optimal airspeed cost $J_i^{\overline{V}*}$, and the corresponding optimal bank angle cost $J_i^{\phi*}$ by assigning termination costs of zero to all final states, and costs of infinity to all states that are not final states.

if $\overline{V}_i \in [\overline{V}_{f_{\min}}, \overline{V}_{f_{\max}}]$ **and** $\gamma_i = 0$ **and** $\phi_{Wi} = 0$ **then**
$J_i^{h*} \leftarrow 0$
$J_i^{\overline{V}*} \leftarrow 0$
$J_i^{\phi*} \leftarrow 0$
else
$J_i^{h*} \leftarrow \infty$
$J_i^{\overline{V}*} \leftarrow \infty$
$J_i^{\phi*} \leftarrow \infty$
end if

6. Populate the optimal next state array $\mathbf{j}^*$ so that the optimal next state (index j) for all final states is the state itself (index i), and that the optimal next state for all states that are not final states is unknown (infinity).

if $\overline{V}_i \in [\overline{V}_{f_{\min}}, \overline{V}_{f_{\max}}]$ **and** $\gamma_i = 0$ **and** $\phi_{Wi} = 0$ **then**
$j_i^* \leftarrow i$
else
$j_i^* \leftarrow \infty$
end if

7. Populate the optimal control array $\mathbf{U}^*$ so that the optimal control for each final state is the trim angle of attack and thrust for that state, and that the optimal control for all states that are not final states is unknown (infinity).

if $\overline{V}_i \in [\overline{V}_{f_{\min}}, \overline{V}_{f_{\max}}]$ **and** $\gamma_i = 0$ **and** $\phi_{Wi} = 0$ **then**
$\mathbf{u}_i^* \leftarrow \mathbf{u}_{i,\text{trim}}$
else
$\mathbf{u}_i^* \leftarrow \infty$
end if

where the trim control input vector $\mathbf{u}_{i,\text{trim}}$ contains the trim angle of attack $\alpha_{i,\text{trim}}$, the trim thrust $T_{i,\text{trim}}$ and the trim wind-axis roll rate $P_{Wi,\text{trim}}$

$$\mathbf{u}_{i,\text{trim}} = \begin{bmatrix} \alpha_{i,\text{trim}} & P_{Wi,\text{trim}} & T_{i,\text{trim}} \end{bmatrix}^T \tag{7.96}$$

which are calculated for each airspeed $\overline{V}_i$ using the following equations

$$\alpha_{i,\text{trim}} = C_L^{-1}\left(\frac{mg}{\frac{1}{2}\rho\overline{V}_i^2 S}\right) \tag{7.97}$$

$$T_{i,\text{trim}} = \frac{1}{2}\rho\overline{V}_i^2 S C_D(\alpha_{i,\text{trim}}) \tag{7.98}$$

$$P_{Wi,\text{trim}} = 0 \tag{7.99}$$

In other word, the trim angle of attack is calculated to produces lift equal to the gravitational force, and the trim thrust is calculated to equal the drag at the given airspeed.

Algorithm Execution

The execution of the dynamic programming algorithm to produce the lookup table of optimal state trajectories and optimal control inputs is now described.

1. Step one time instant backward in time

   $$k \leftarrow k - 1$$

2. Set the index i of the current state $\mathbf{x}_i(k)$ and the index j of the next state $\mathbf{x}_j(k)$ to the first index of the state vector array

   $$i \leftarrow 1$$
   $$j \leftarrow 1$$

3. For the given index j of the next state $\mathbf{x}_j(k+1)$, iterate through all possible values of the current state $\mathbf{x}_i(k)$ by iterating through the current state index i. For each possible next state, do the following:

4. First check the optimal altitude cost of the next state $J_j^{h*}(k+1)$. If the optimal altitude cost from the next state is infinite, it means that an admissible trajectory has not been found from the next state to a final state yet, and we should therefore not transition to this next state (increment the next state index j and return to step 3). However, if the optimal altitude cost from the next state is finite, then we can attempt to transition from the current state to this next state (proceed to the next step).

5. Calculate the control input $\mathbf{u}_i$ required to transition from the current state $\mathbf{x}_i(k)$ to the candidate next state $\mathbf{x}_j(k+1)$

   $$\mathbf{u}_i(k) = \begin{bmatrix} \alpha_i & P_{Wi}(k) & T_i(k) \end{bmatrix}^T \tag{7.100}$$

   where the required angle of attack, roll rate, and thrust are calculated with

   $$\begin{aligned} \alpha_i(k) &\leftarrow C_L^{-1}\left(\frac{m\overline{V}_i(k)\frac{\gamma_j(k+1)-\gamma_i(k)}{\Delta t} + mg\cos\gamma_i(k)}{\frac{1}{2}\rho\overline{V}_i^2(k)S\cos\phi_{Wi}(k)}\right) \\ T_i(k) &\leftarrow \frac{m[\overline{V}_j(k+1) - \overline{V}_i(k)]}{\Delta t} + \frac{1}{2}\rho\overline{V}_i^2(k)SC_D(\alpha_i(k)) + mg\sin\gamma_i(k) \\ P_{Wi}(k) &\leftarrow \frac{\phi_{Wj}(k+1) - \phi_{Wi}(k)}{\Delta t} \end{aligned}$$

6. Check whether the control input $\mathbf{u}_i$ is an admissible input. If

   $$\begin{aligned} \alpha_i(k) &\in [\alpha_{\min}, \alpha_{\max}] \\ \text{and } P_{Wi}(k) &\in [-P_{W\max}, P_{W\max}] \\ \text{and } T_i(k) &\in [T_{\min}, T_{\max}] \\ \text{and } n_{Li}(k) &\in [n_{L\min}, n_{L\max}] \end{aligned}$$

   then the control input $\mathbf{u}_i(k)$ is admissible, otherwise it is not. If the control input is admissible, then proceed to the next step, otherwise jump to step 9.

7. Calculate the hierarchical cost from the current state $\mathbf{x}_i(k)$ to a final state through this next state $\mathbf{x}_j(k+1)$

$$\begin{aligned} J_i^h &\leftarrow \Delta J_{ij}^h + J_j^{h*} \\ J_i^{\overline{V}} &\leftarrow \Delta J_{ij}^{\overline{V}} + J_j^{\overline{V}*} \\ J_i^{\phi} &\leftarrow \Delta J_{ij}^{\phi} + J_j^{\phi*} \end{aligned}$$

8. If the new hierarchical cost is lower than the lowest hierarchical cost so far, it becomes the new lowest hierarchical cost. In other words, if $(J_i^h < J_i^{h*})$ or $(J_i^h = J_i^{h*}$ and $J_i^{\overline{V}} < J_i^{\overline{V}*})$ or $(J_i^h = J_i^{h*}$ and $J_i^{\overline{V}} = J_i^{\overline{V}*}$ and $J_i^{\phi} < J_i^{\phi*})$ then

$$\begin{aligned} J_i^{h*} &\leftarrow J_i^h \\ J_i^{\overline{V}*} &\leftarrow J_i^{\overline{V}} \\ J_i^{\phi*} &\leftarrow J_i^{\phi} \end{aligned}$$

Also, the control input $\mathbf{u}_i(k)$ and the next state index j become the new best control input $\mathbf{u}_i^*(k)$ and the new best next state j_i^* for the current state $\mathbf{x}_i(k)$.

$$\begin{aligned} \mathbf{u}_i^* &\leftarrow \mathbf{u}_i(k) \\ j_i^* &\leftarrow j \end{aligned}$$

9. Increment the current state index i. If the current state index i was the last index, then increment the next state index j, and reset the current state index i to the first index. If the next state index j was the last index, then step one time instant k backward, and reset the next state index j to the first index. Return to step 3.

Finalisation

The dynamic programming algorithm is terminated when one of the following conditions are met:

- All the states in the state vector array have been assigned finite optimal costs. (This means that optimal and admissible trajectories have been found from all initial states to a final state.)
- No new state transitions have been added for an entire iteration of the state vector array. This implies that further iterations would produce the same results and no further state transitions will be added. (This means that no admissible trajectories exist for the states that do not have finite costs assigned to them.)
- The maximum number of time steps has been reached.

After the dynamic programming algorithm has completed its search, all states with finite costs are considered "recoverable" and have optimal and admissible state trajectories and control sequences to a final state, and all states with infinite costs are considered "unrecoverable" because no admissible trajectories or control sequences exist from these states to a final state.

Search Optimisations

The following search optimisations have been implemented

1. A check was added so that the search algorithm only considers transitions *to* states that already have an admissible path to a final state. When iterating through the state vector array at a given time instant k, the search algorithm only considers transitions *to* states at the next time instant $\mathbf{x}_j(k+1)$ that have finite costs associated with them, i.e. for which J_j^{h*} is finite. If a state $\mathbf{x}_j$ has an infinite cost associated with it, it means that an admissible path from that state to a final state has not been found yet (and possible does not exist), and it therefore does not make sense to transition to it. In other words, the search algorithm does not try to find transitions to states that do not themselves have an admissible path to a final state. In this way, the algorithm does not waste computational effort checking the admissibility and cost of the state transition.

2. A check was added so that the search algorithm only considers transitions *to* states for which their total path costs were updated in the previous complete time step iteration. If the total path cost associated with a state was updated in the previous complete time step iteration, then all states must check again in this time step iteration if transitioning to the updated state would result in a lower total path cost for themselves. However, if the total path cost of a next state was not updated in the previous complete time step iteration, then all states have already checked the admissibility and the total path cost of transitioning to that state in a previous iteration, and checking it again would just produce the same result. In this way, the algorithm does not waste computational effort repeating calculations for state transitions that have already been performed and would only produce the same result again.

3. The symmetry of the bank angle recovery is exploited so that the dynamic programming search algorithm only needs to consider the positive half of the bank angles search space. The lookup table therefore only contains states with positive bank angles and negative roll rates. When the lookup table is navigated, negative bank angles are treated as their positive counterparts, and the negative roll rates obtained from the lookup table are treated as positive roll rates. In this way, the size of the quantised state vector array is halved.

Lookup Table Navigation

The dynamic programming algorithm produces a lookup table of the optimal state trajectories and optimal control sequences for recovery from all admissible initial states. Given an initial state $\mathbf{x}_i(1)$, the optimal state trajectory $\{\mathbf{x}^*(k) : k = 1, 2, \ldots, N\}$ and the optimal control sequence $\{\mathbf{u}^*(k) : k = 1, 2, \ldots, N\}$ can be obtained by starting at the index i of the initial state and iteratively navigating through the lookup table by following the optimal next state indexes j^*.

$k \leftarrow 1$
$i \leftarrow i_0$
while $j* \neq i$ **and** $k \leq N$ **do**
  $\mathbf{x}^*(k) = \mathbf{x}_i$
  $\mathbf{u}^*(k) = \mathbf{u}_i$
  $i \leftarrow j_i^*$
  $k \leftarrow k + 1$
end while

The maximum altitude that will be lost and the maximum airspeed that will be reached during the recovery from a given initial state $\mathbf{x}(0)$ can be obtained directly from the lookup table by reading the optimal altitude cost J_i^{h*} and the optimal airspeed cost $J_i^{\overline{V}^*}$ at the index i_0 of the initial state. The lookup table therefore predicts how much altitude would be lost and what the maximum speed is that will be reached if the recovery procedure is activated from the given state.

For initial states that are between grid points, the lookup table can be navigated by using either a nearest neighbour interpolation, or a multilinear interpolation. When using nearest neighbour interpolation, we use the optimal control input $\mathbf{u}_i^*$ of the quantised state $\mathbf{x}_i$ nearest to the actual initial state $\mathbf{x}$. When using multilinear interpolation, we obtain the optimal control input by performing multilinear interpolation of the optimal control inputs of all the quantised states neighbouring the actual state. In both cases, we use the optimal control input and the discrete-time difference equations to propagate the *actual* state to the next time instant. At the next time instant, we obtain the optimal control for the propogated actual state using nearest neigbour or multilinear interpolation again, and repeat.

7.7 Dynamic Programming Results

This section presents the results of the dynamic programming algorithm using the physical parameters and aerodynamic functions of the NASA Generic Transport Model. First, the simulation setup and the physical parameters and aerodynamic functions used are discussed. Next, the comprehensive set of optimal state trajectories and control inputs from all recoverable initial states are presented, accompanied by comprehensive maps of all recoverable and unrecoverable initial states. Finally, some illustrative example recovery trajectories are shown and discussed. The examples illustrate how the dynamic programming algorithm performs optimal bank angle, flight path angle, and airspeed recovery from various upset scenarios that include combinations of unusual bank angle, unusual flight path angle, underspeed and overspeed initial conditions.

7.7.1 Dynamic Programming Setup

The constants and parameter values shown in table 7.1 were used for the dynamic programming. The aerodynamic lift and drag coefficient functions $C_L(\alpha)$ and $C_D(\alpha)$ with angle of attack α as the input argument are shown in figure 7.5. Note that only the sections of functions where the angle of attack is between 0 and 21 degrees will be used, since the elevators cannot produce enough pitching moment to maintain angles of attack above 21 degrees.

The range of admissible airspeeds and the range of admissible final airspeeds have been selected based on the trimming curves of the NASA GTM. The minimum final airspeed has been selected as 70 knots, since the trim angle of attack increases rapidly at lower airspeeds. The maximum final airspeed has been selected as 100 knots, since the trim throttle increases above 30% at higher airspeeds. The minimum admissible airspeed as been selected as 30 knots, since it is well below the stall airspeed. The maximum admissible airspeed has arbitrarily been selected as 140 knots, since it is well above the maximum acceptable airspeed.

To make things more interesting, a fictitious roll rate envelope has been added as an additional constraint. The roll rate envelope constrains the maximum roll rate that may be commanded as a function of the angle of attack. The roll rate envelope has been modelled after the roll rate envelope of the X-31 High Angle of Attack Control System [3]. The fictitious roll rate envelope created for the NASA GTM and the X-31 roll rate envelope are both shown in figure 7.6. Up to an angle of attack of 7 degrees, the roll rate is allowed to go up to its normal maximum of 30 degrees per second. As the angle of attack increases beyond 7 degrees, the maximum roll rate command is decreased with an exponential curve down to a maximum of 5 degrees per second at angles of attack beyond 20 degrees.

Table 7.1: Physical constants and NASA GTM parameters

Parameter	**Symbol**	**Value**	**Units**
Physical constants			
mass	m	1.6162	slug
gravitional acceleration	g	32	ft/s^2
air density	ρ	0.0023769	slug/ft^3
lift coefficient function	$C_L(\alpha)$	figure 7.5	
drag coefficient function	$C_D(\alpha)$	figure 7.5	
engine time constant	τ	2.36	s
Admissible states			
airspeed	$[\overline{V}_{\min}, \overline{V}_{\max}]$	$[5, 140]$	knots
airspeed, final	$[\overline{V}_{f\min}, \overline{V}_{f\max}]$	$[70, 100]$	knots
flight path angle	$[\gamma_{\min}, \gamma_{\max}]$	$[-90, 30]$	deg
bank angle (in wind axes)	$[\phi_{W\min}, \phi_{W\max}]$	$[0, 180]$	deg
Admissible inputs			
angle of attack	$[\alpha_{\min}, \alpha_{\max}]$	$[-5, 21]$	deg
engine thrust	$[T_{\min}, T_{\max}]$	$[0, 15]$	lbs
roll rate (in wind axes)	$[P_{W\min}, P_{W\max}]$	$[-30, 30]$	deg/s
Additional constraints			
load factor	$[n_{z\min}, n_{z\max}]$	$[-1, 2.5]$	g
roll rate envelope	$P_{W\max}(\alpha)$	figure 7.6	deg/s

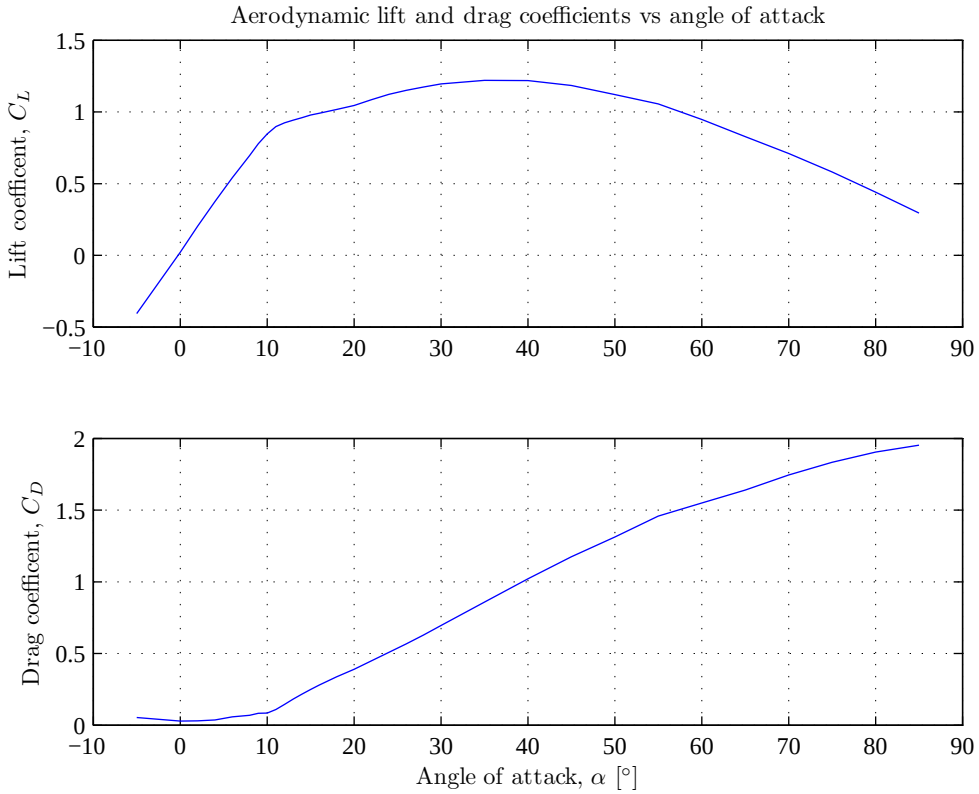


Figure 7.5: Aerodynamic lift and drag coefficient functions

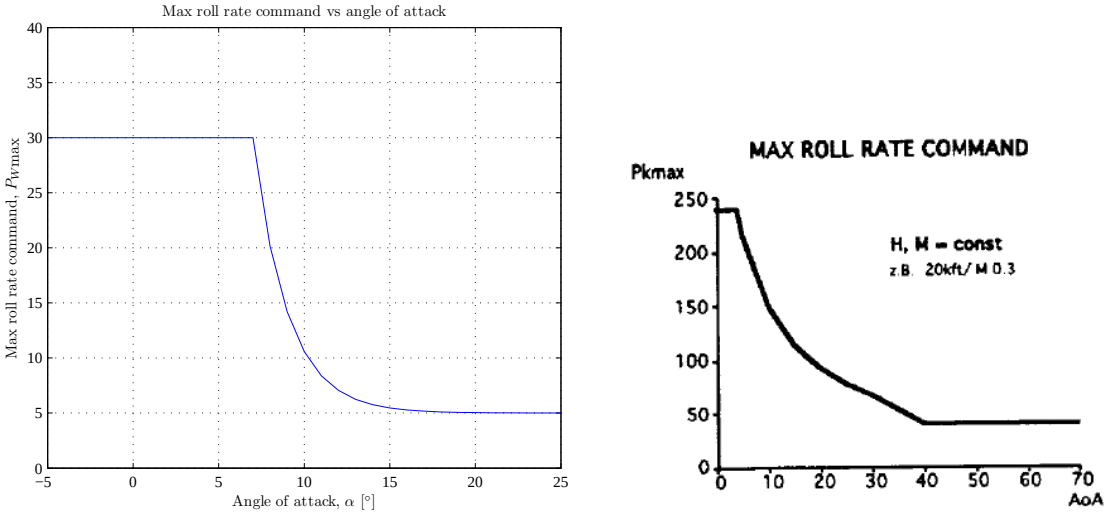


Figure 7.6: Max roll rate envelope vs. angle of attack (modelled after [3])

7.7.2 Optimal State Trajectories and Control Inputs

The comprehensive set of optimal state trajectories and control inputs for all recoverable initial states on the state grid are shown in figures 7.7 and 7.8 respectively. The state trajectories plots show the time histories of the airspeed, flight path angle, and wind-axis bank angle. The control input trajectories plots show the time histories of the angle of attack command and the wind axis roll rate command, with the thrust assumed constant. Also shown in figure 7.8 are the time histories of the normal load factor experienced, to verify that it remains within safe limits. The altitude trajectories during the optimal recovery maneuvers are shown in figure 7.9. These results were generated by navigating the dynamic programming lookup tables from all recoverable initial states.

The optimal state trajectories in figure 7.7 show that the aircraft is recoverable for airspeeds from 20 knots to 140 knots, flight path angles from -90 to +30 degrees, and bank angles from -180 to +180 degrees. The duration of the recovery may be as short as one second, or could take up to 19 seconds for the most severe upset conditions. In all cases, the aircraft is recovered to straight and level flight (zero flight path angle, zero bank angle) with final airspeeds ranging from 70 to 100 knots.

The optimal control inputs trajectories in figure 7.8 show that angle of attack commands from 0 to 21 degrees are used to control and recover the flight path angle, and that wind-axis roll rate commands from -30 to +30 degrees per second are used to control and recover the bank angle. The thrust command is kept constant during the recovery maneuver, and is only adjusted in the last step, to set it to the trim thrust required to maintain the final airspeed. During the recovery maneuver, the flight path angle is used to control the airspeed, instead of the thrust. The airspeed is controlled and recovered by using negative flight path angles to increase the airspeed, and positive flight path angles to reduce the airspeed. For all recovery trajectories, the load factor ranges between 0 and 2.5g, and therefore remains safely within the structural integrity envelope.

The optimal altitude trajectories in figure 7.9 show that for most trajectories the aircraft loses between 0 and 1000 feet of altitude, but for some trajectories the aircraft actually gains up to 430 feet of altitude. The reason that the aircraft sometimes gains altitude, is because it uses a positive flight path angle to reduce the airspeed. The trajectories that have a net gain in altitude typically include a component of overspeed recovery. It is also observed that for some trajectories the peak altitude loss is greater than the net altitude loss.

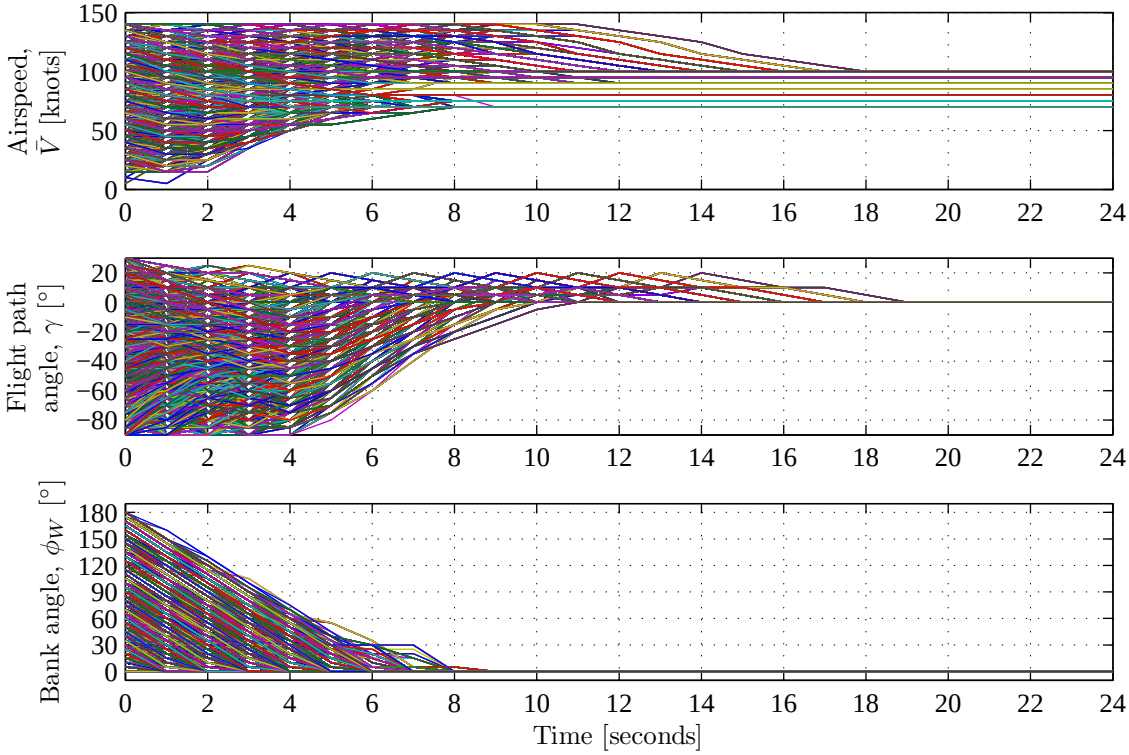


Figure 7.7: Optimal state trajectories from all recoverable states

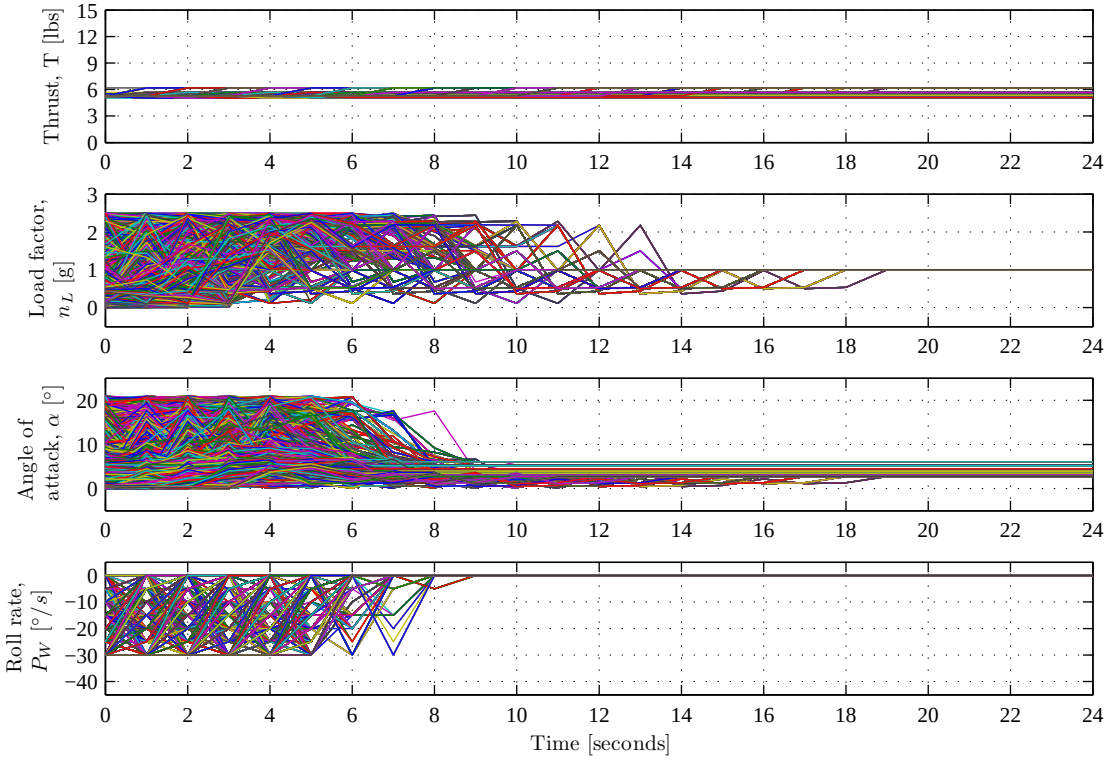


Figure 7.8: Optimal control input trajectories from all recoverable states

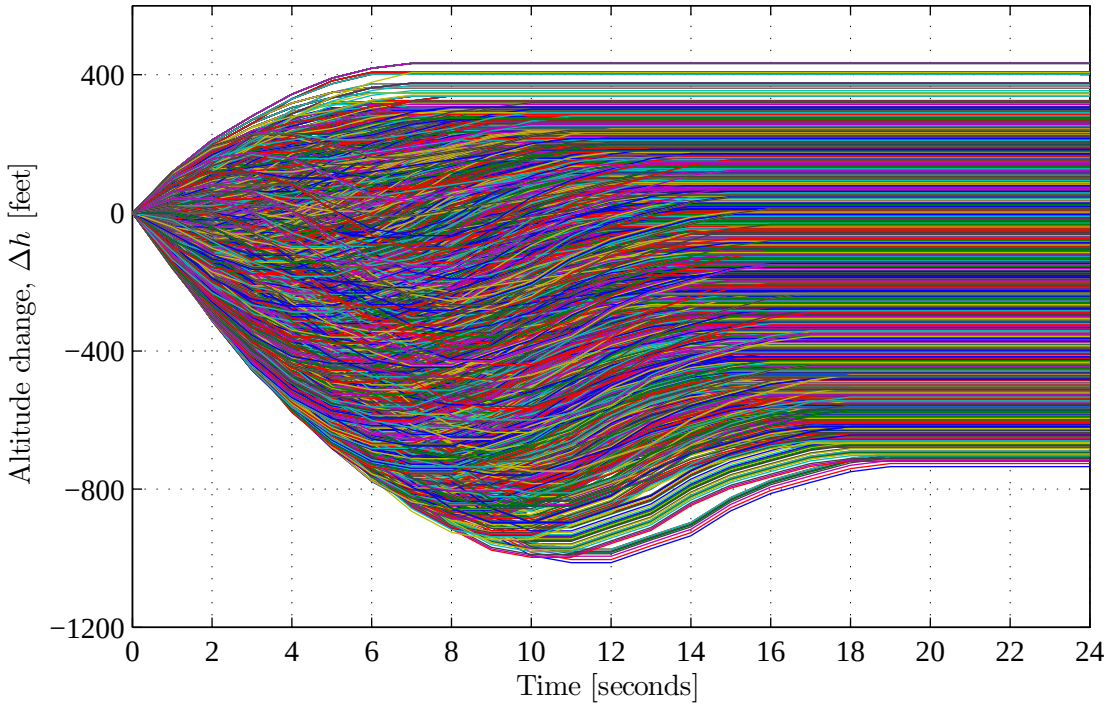


Figure 7.9: Optimal altitude trajectories from all recoverable states

There is an initial "dip" in the altitude trajectory, and then the aircraft recovers some altitude before it settles to a constant final altitude. This happens when the aircraft enters overspeed while recovering the bank angle and the flight path angle, and then needs to use a positive flight path angle to recover from the overspeed condition.

The cumulative distribution of the altitude lost over all optimal recovery trajectories is shown in figure 7.10. The distribution shows that 28% of the optimal recovery trajectories lose less than 100 feet of altitude, 75% of the trajectories lose less than 400 feet, 92% of the trajectories lose less than 600 feet, and none of the trajectories lose more than 1000 feet.

The cumulative distribution of the maximum airspeed reached over all optimal recovery trajectories is shown in figure 7.11. The distribution shows that 64% of the trajectories do not exceed 100 knots, but that 36% of the trajectories do enter admissible overspeed conditions. However, it must be remembered that the 36% includes trajectories that are already initialised with an overspeed airspeed ranging from 100 to 140 knots.

7.7.3 Recoverable and Unrecoverable States

Unfortunately, the aircraft can not be recovered from all initial airspeeds, flight path angles, and bank angles without exceeding the structural integrity envelope of the aircraft. This is due to the physical and aerodynamic limitations of the aircraft, and not through any fault of the dynamic programming algorithm. In fact, since the dynamic programming algorithm finds the optimal trajectories, it reveals from which initial states recovery is possible, and from which initial states recovery is not possible without exceeding the normal load factor or maximum airspeed limitations.

Maps of the recoverable and unrecoverable initial states are shown in figures 7.12 and 7.13. The information about which states are recoverable and which states are unrecoverable is explicitly contained in the lookup table that stores the optimal recovery cost from each state to a final state that is generated by the dynamic programming algorithm.

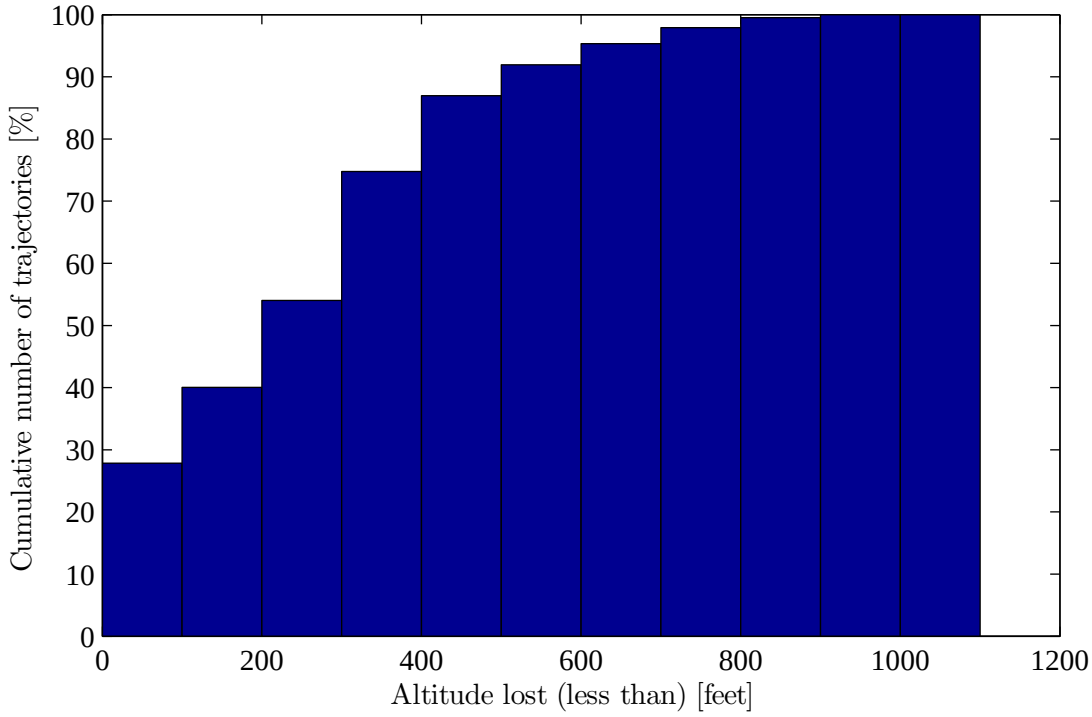


Figure 7.10: Cumulative distribution of altitude lost during recovery. Cumulative histogram generated from the complete set of 207885 recoverable trajectories.

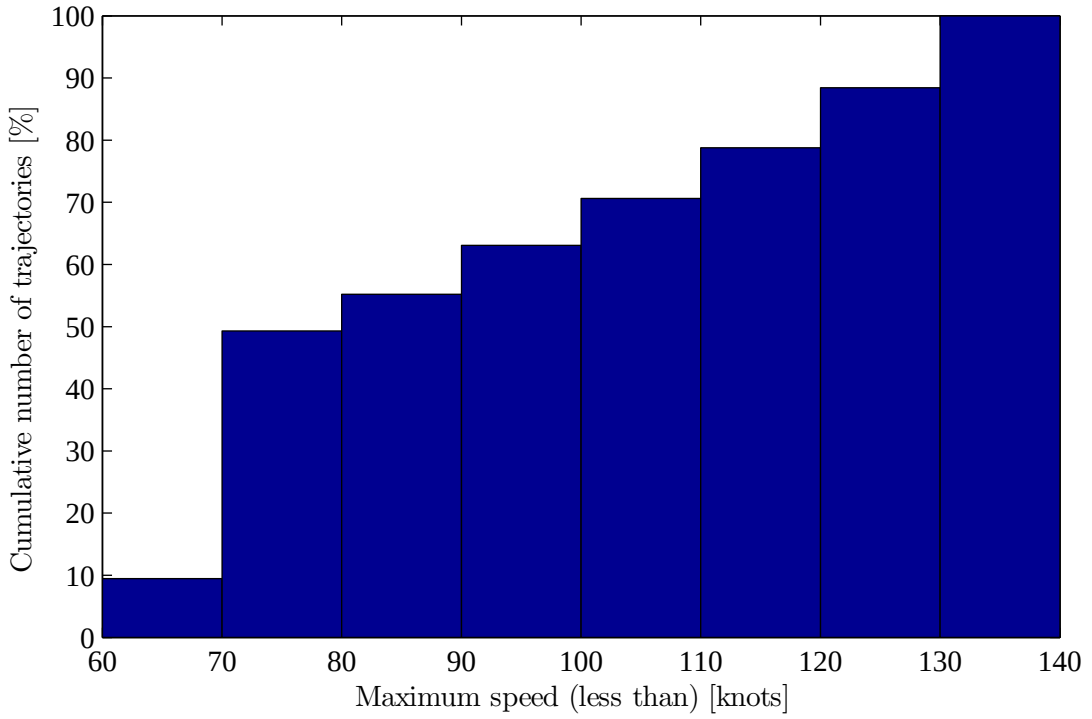


Figure 7.11: Cumulative distribution of maximum airspeed reached during recovery. Cumulative histogram generated from the complete set of 207885 recoverable trajectories.

If the optimal cost associated with a state is finite, then the state is recoverable and the optimal cost represents the maximum altitude that will be lost during the recovery maneuver. If the optimal cost associated with a state is infinite, then the state is unrecoverable, which means that the aircraft cannot be recovered to straight and level flight at an acceptable airspeed, without exceeding the load factor limits or the absolute maximum airspeed of the aircraft.

Figure 7.12 shows the map of recoverable and unrecoverable states for all "longitudinal" recovery trajectories, where the aircraft is at least initially wings level (zero bank angle). The map shows the recoverable and unrecoverable states for all combinations of airspeed and flight path angle with the airspeed ranging from 5 to 140 knots, and the flight path angle ranging from -90 to +30 degrees. The map shows two distinct unrecoverable regions, namely an "unrecoverable underspeed" region, and an "unrecoverable descending overspeed" region. The aircraft cannot be recovered from states where the airspeed is too low (less than 15-20 knots), regardless of the flight path angle, and the aircraft can also not be recovered from states with high airspeeds combined with large negative flight path angles (a triangular region of airspeeds greater than 100 knots and flight path angles less than -25 degrees). At high airspeeds the flight path angle rate is limited by the load factor constraints, and for large negative flight path angles the flight path angle cannot be recovered quickly enough before the aircraft exceeds its maximum airspeed. For very low initial airspeeds, the flight path angle rapidly drops while the airspeed steadily increases under the influence of gravity. This leads to a combined high airspeed and large negative flight path angle from which the aircraft cannot be recovered.

Figure 7.13 shows the maps of recoverable and unrecoverable states for "three-dimensional" recovery trajectories for combinations of airspeed, flight path angle, and bank angle. The maps show the recoverable and unrecoverable states for all combinations of flight path angle and bank angle, over a range of airspeeds, with the flight path angle ranging from -90 to +30 degrees, the bank angle ranging from 0 to 180 degrees, and the airspeed ranging from 30 to 140 knots.

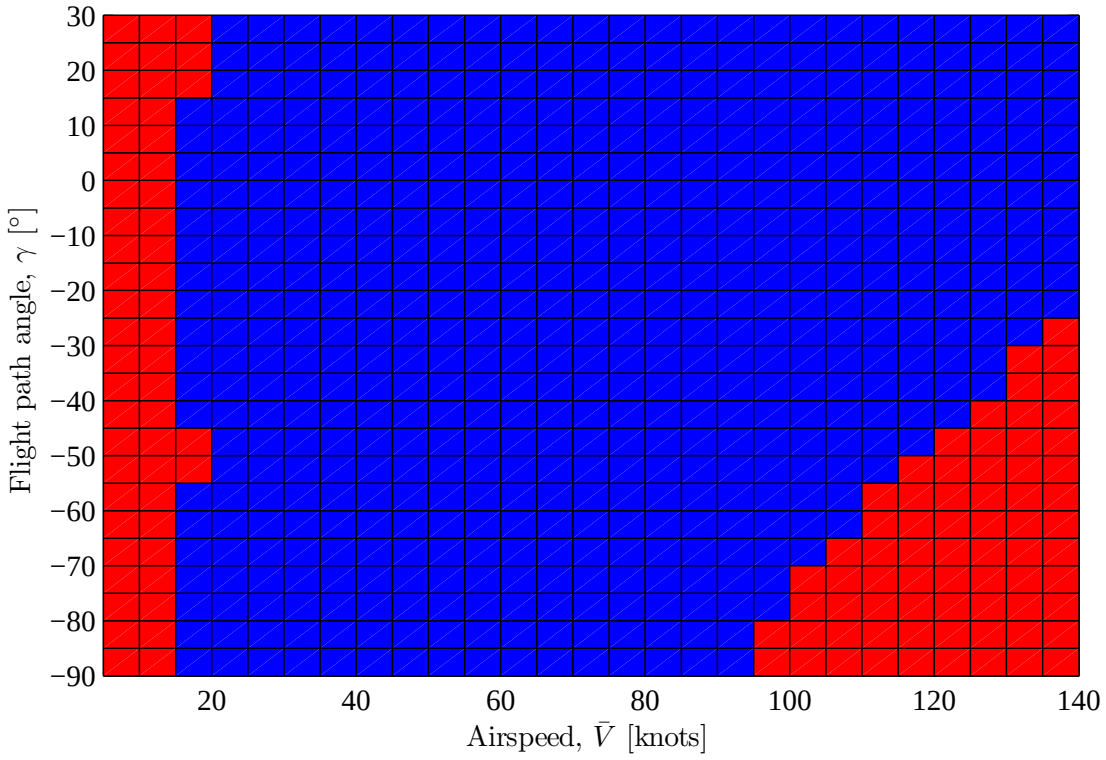


Figure 7.12: Map of recoverable and unrecoverable states. Airspeed vs. flight path angle, zero bank angle. (blue = recoverable, red = unrecoverable)

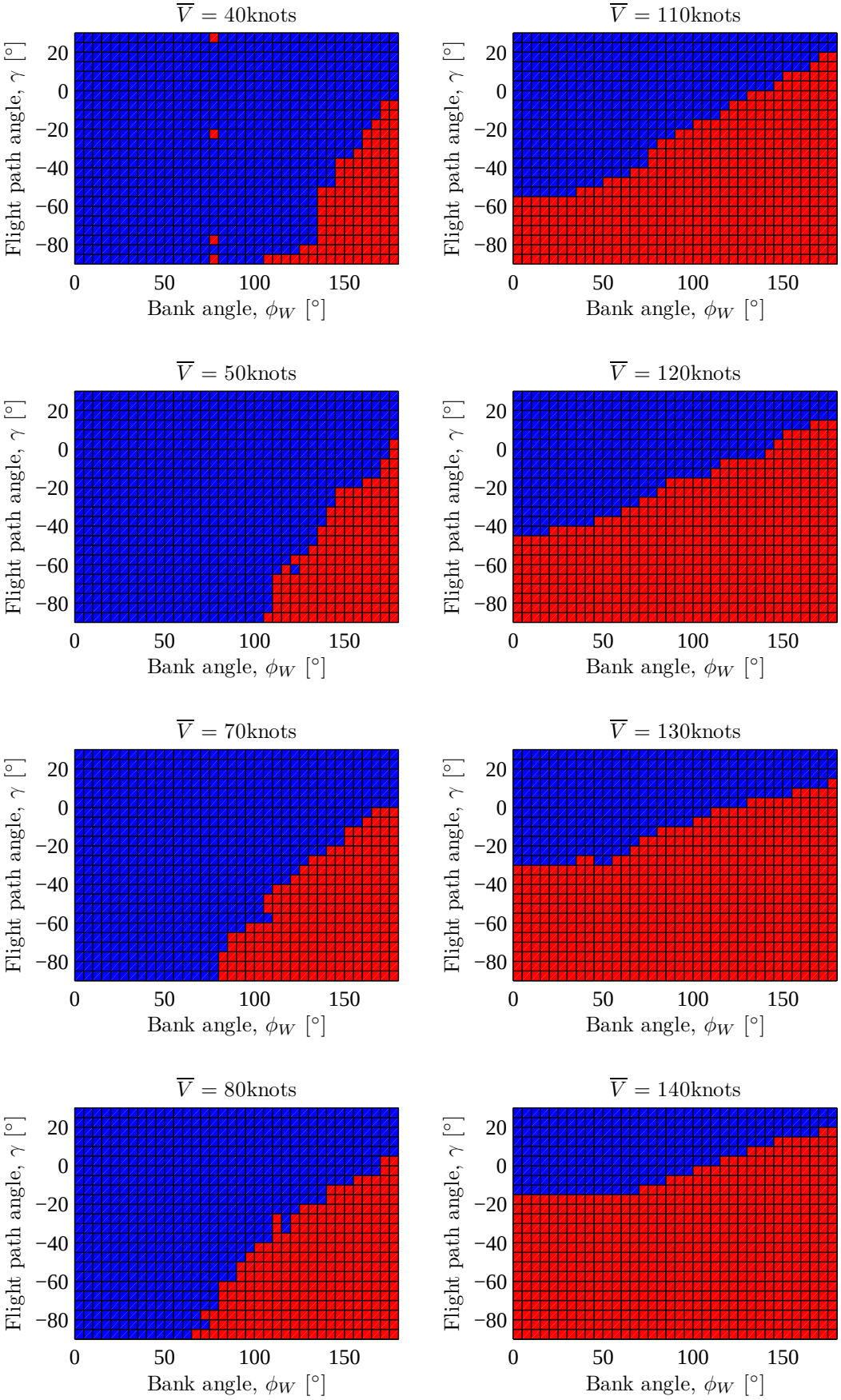


Figure 7.13: Maps of recoverable and unrecoverable states. Flight path angle vs. bank angle, at various airspeeds. (blue = recoverable, red = unrecoverable)

At lower airspeeds (40 to 80 knots), the region of recoverable states is larger, and the aircraft can be recovered from steeply descending flight path angles, but with low bank angles, and from high or even inverted bank angles, but with shallow descending or climbing flight path angles. However, the aircraft cannot be recovered from states that have steeply descending flight path angles combined with high or inverted bank angles.

At higher airspeeds (above 110 knots) the region of recoverable states shrinks, and high bank angles can only be recovered for shallower descending flight path angles. As the airspeed increases, the region of recoverable descending flight path angles becomes increasingly shallow. However, the aircraft can still be recovered from an inverted bank angle at high airspeed, as long as the flight path angle is zero or positive.

Some isolated unrecoverable states are observed in the map of recoverable and unrecoverable states for an airspeed of $\overline{V} = 40$ knots. These isolated unrecoverable states all occur at bank angles of 90 degrees, where the lift vector cannot affect the flight path angle because its direction is perpendicular to the vertical. This is because the dynamic programming algorithm could not find an admissible state to which it could transition from this state. Remember that for bank angles of 90 degrees, the algorithm checks if the state transition is admissible by propagating the flight path angle forward and checking whether the propagated flight path angle is within a given tolerance of the flight path angle of the next state. These isolated unrecoverable states should actually be recoverable, and it should be possible to "connect" them to recoverable states by relaxing the tolerance of the state transition admissibility check.

7.7.4 Example Recovery Scenarios

This section presents and discusses the optimal recovery trajectories and control inputs for some example upset scenarios. The examples illustrate how the dynamic programming algorithm performs optimal bank angle, flight path angle, and airspeed recovery from various upset scenarios that include combinations of unusual bank angle, unusual flight path angle, underspeed and overspeed initial conditions. Illustrative results for the following recovery scenarios are presented:

- Flight path angle recovery, steep descent
- Flight path angle recovery, steep climb
- Bank angle recovery
- Underspeed recovery
- Overspeed recovery
- Bank angle and flight path angle recovery, steep descent
- Bank angle and flight path angle recovery, steep descent, overspeed
- Bank angle and flight path angle recovery, steep climb, underspeed

In all cases, it is very pleasing to see that the state trajectories and control inputs produced by the dynamic programming algorithm agree with recovery actions that we would intuitively expect from a human pilot. This is particularly interesting, especially given that the dynamic programming algorithm does not produce a rules-based solution, such as the prescribed upset recovery procedures for pilots, but rather produces a lookup table of optimal recovery trajectories and control inputs.

Flight path angle recovery, steep descent

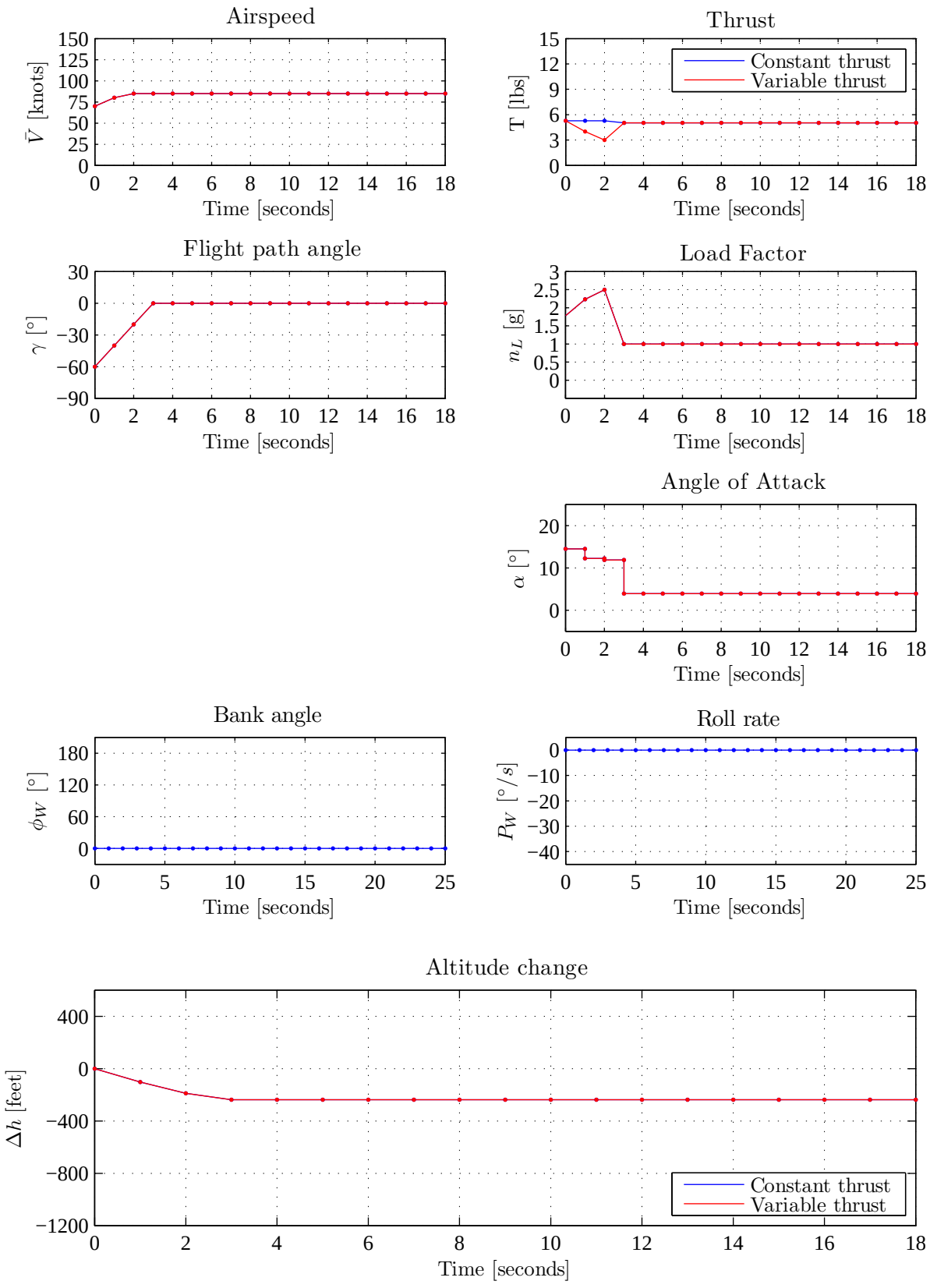


Figure 7.14: Flight path angle recovery: steep descent

Flight path angle recovery, steep climb

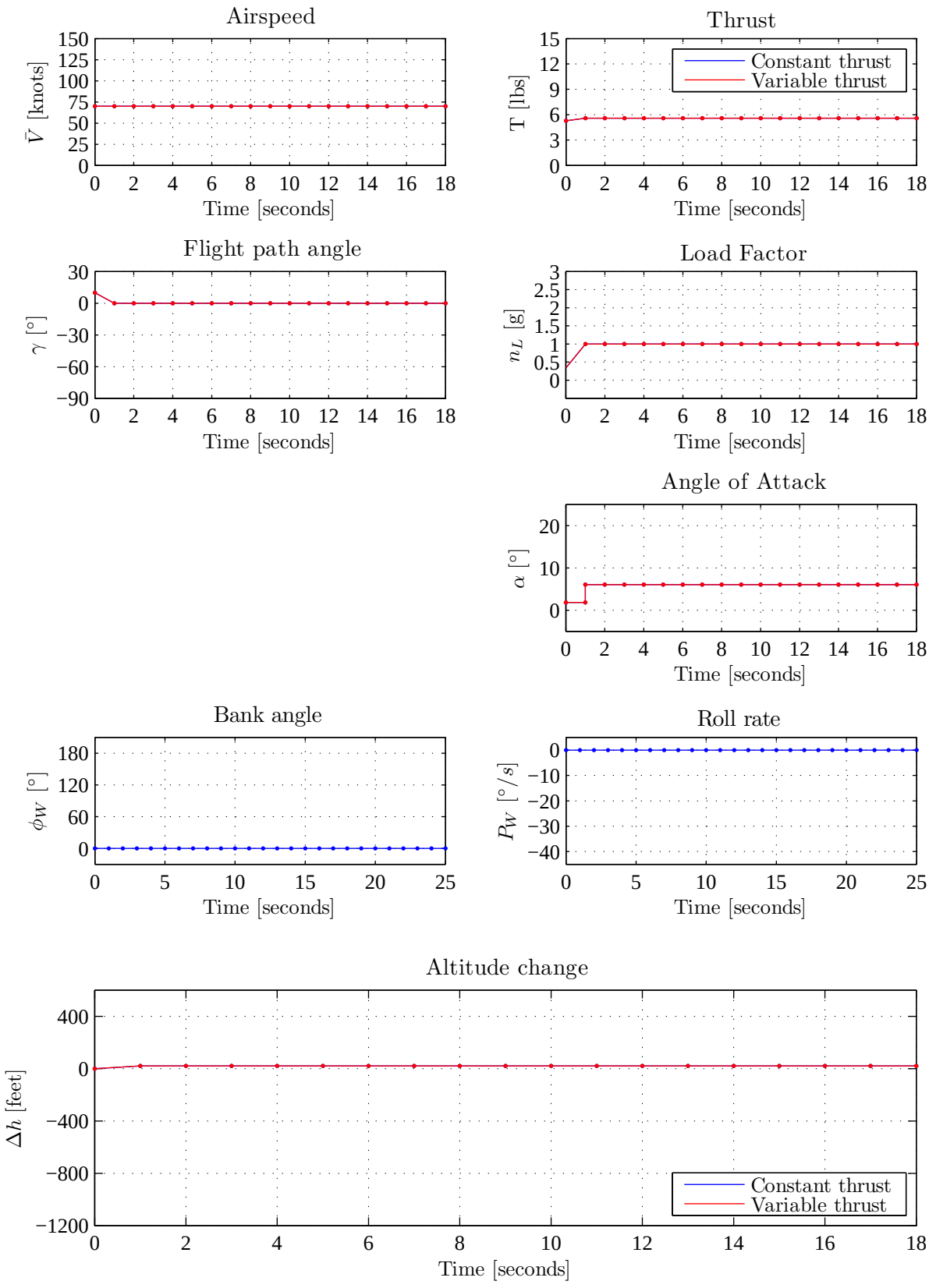


Figure 7.15: Flight path angle recovery: steep climb

Bank angle recovery

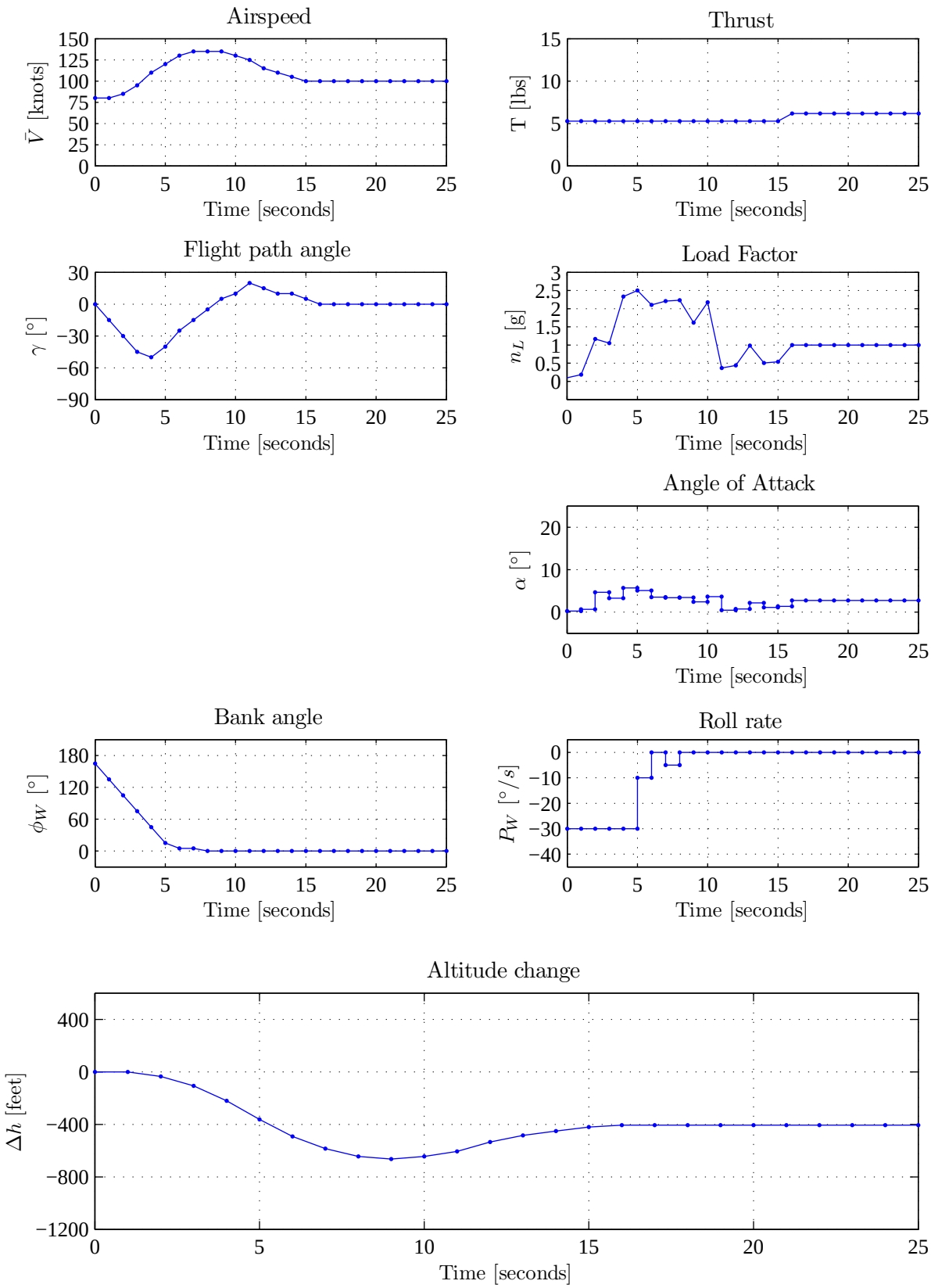


Figure 7.16: Bank angle recovery: aircraft upside-down, level flight path, normal airspeed

Underspeed recovery

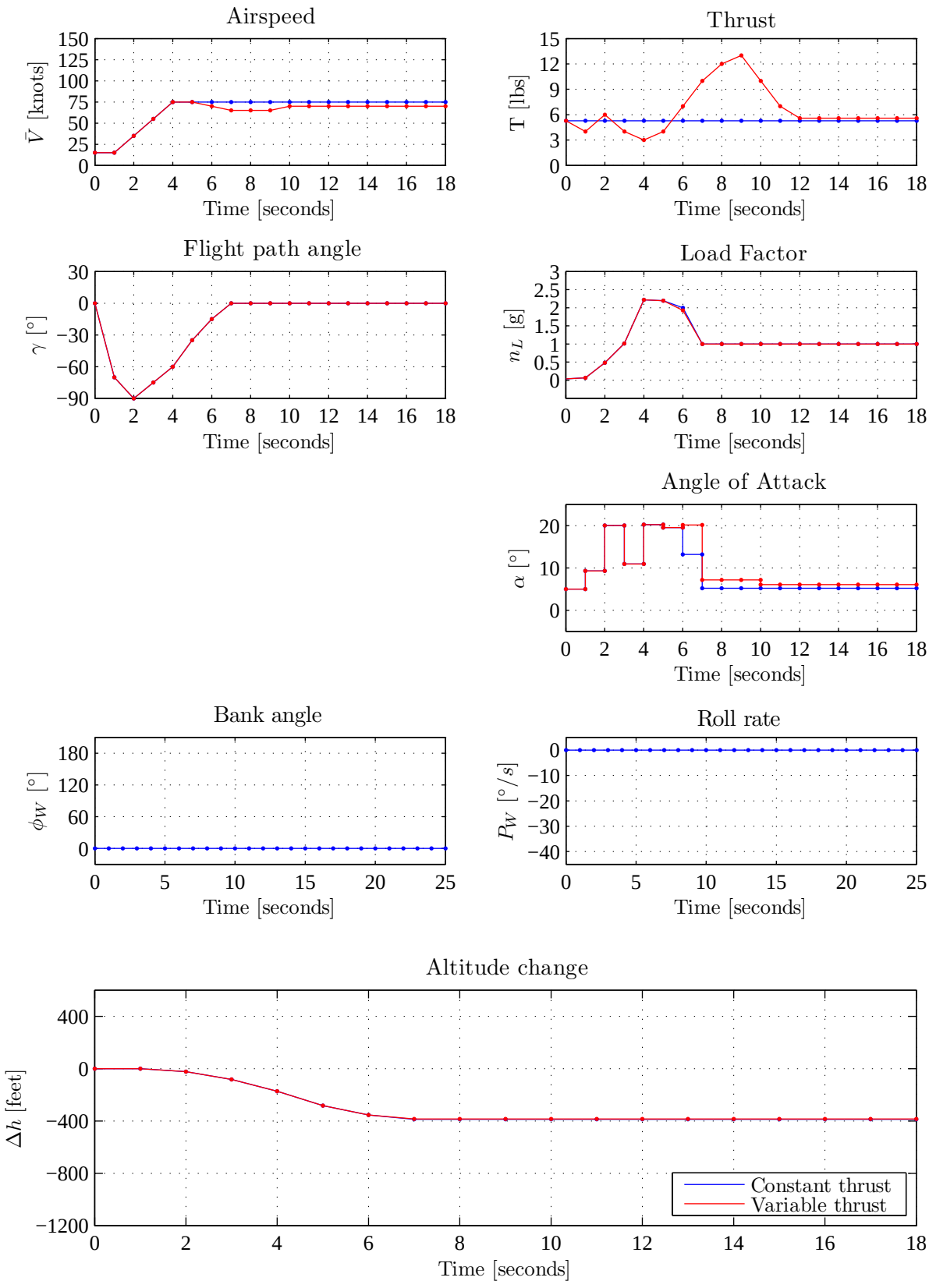


Figure 7.17: Underspeed recovery

Overspeed recovery

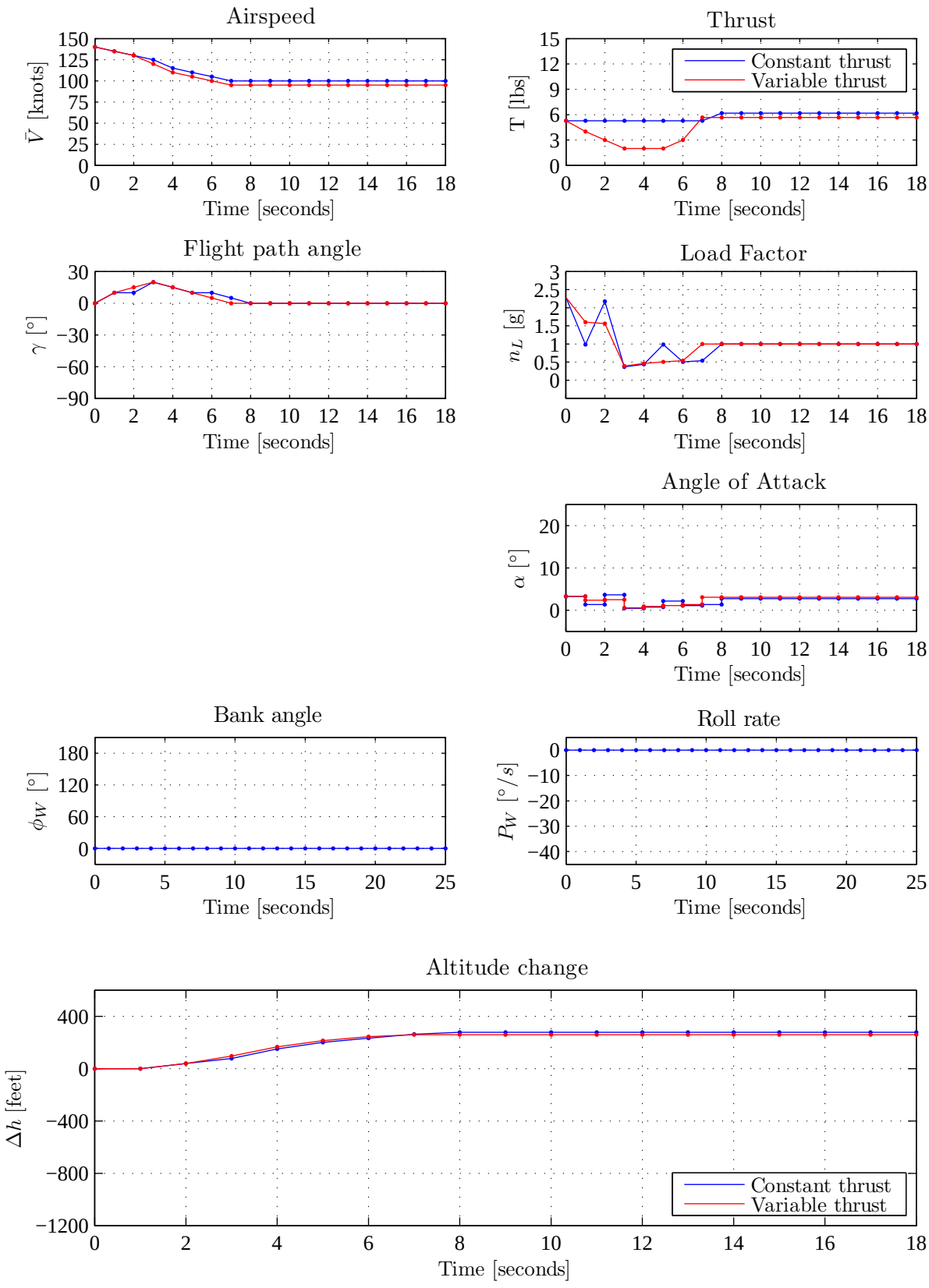


Figure 7.18: Overspeed recovery

Bank angle and flight path angle recovery, steep descent

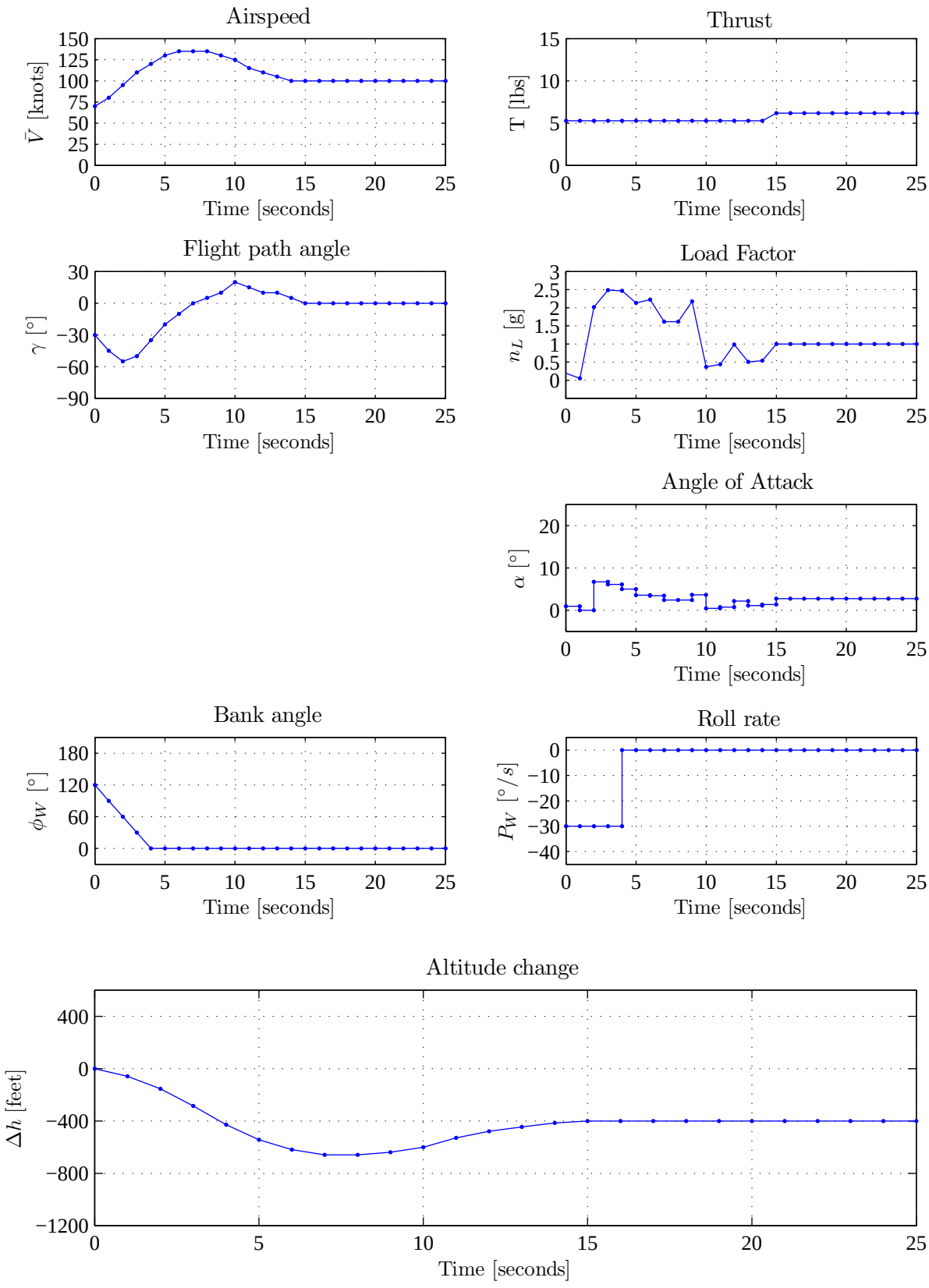


Figure 7.19: Bank angle and flight path recovery: aircraft upside-down, descending flight path, normal airspeed

Bank angle and flight path angle recovery, steep descent, overspeed

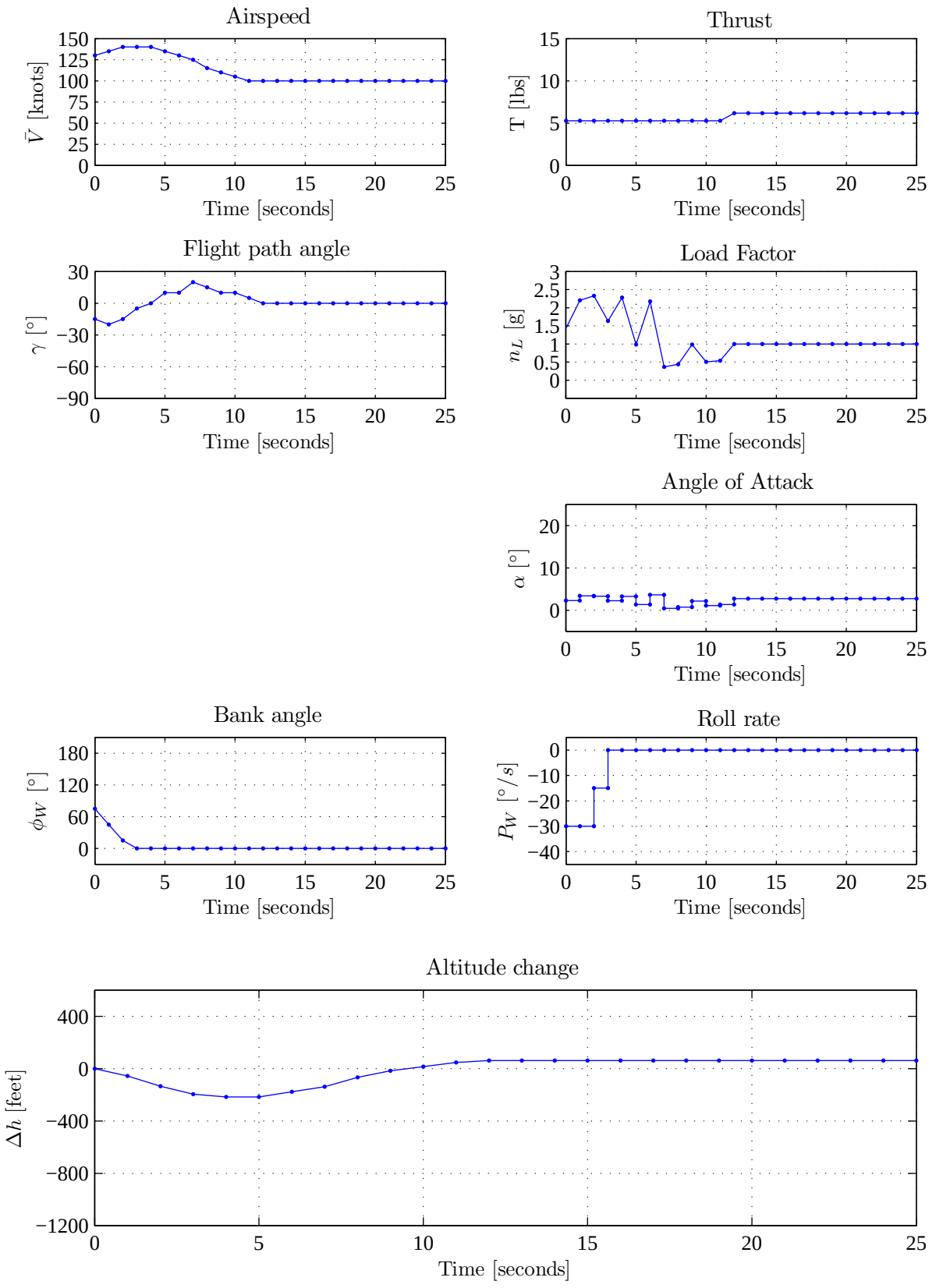


Figure 7.20: Bank angle and flight path recovery: aircraft upside-down, descending flight path, overspeed

Bank angle and flight path angle recovery, steep climb, underspeed

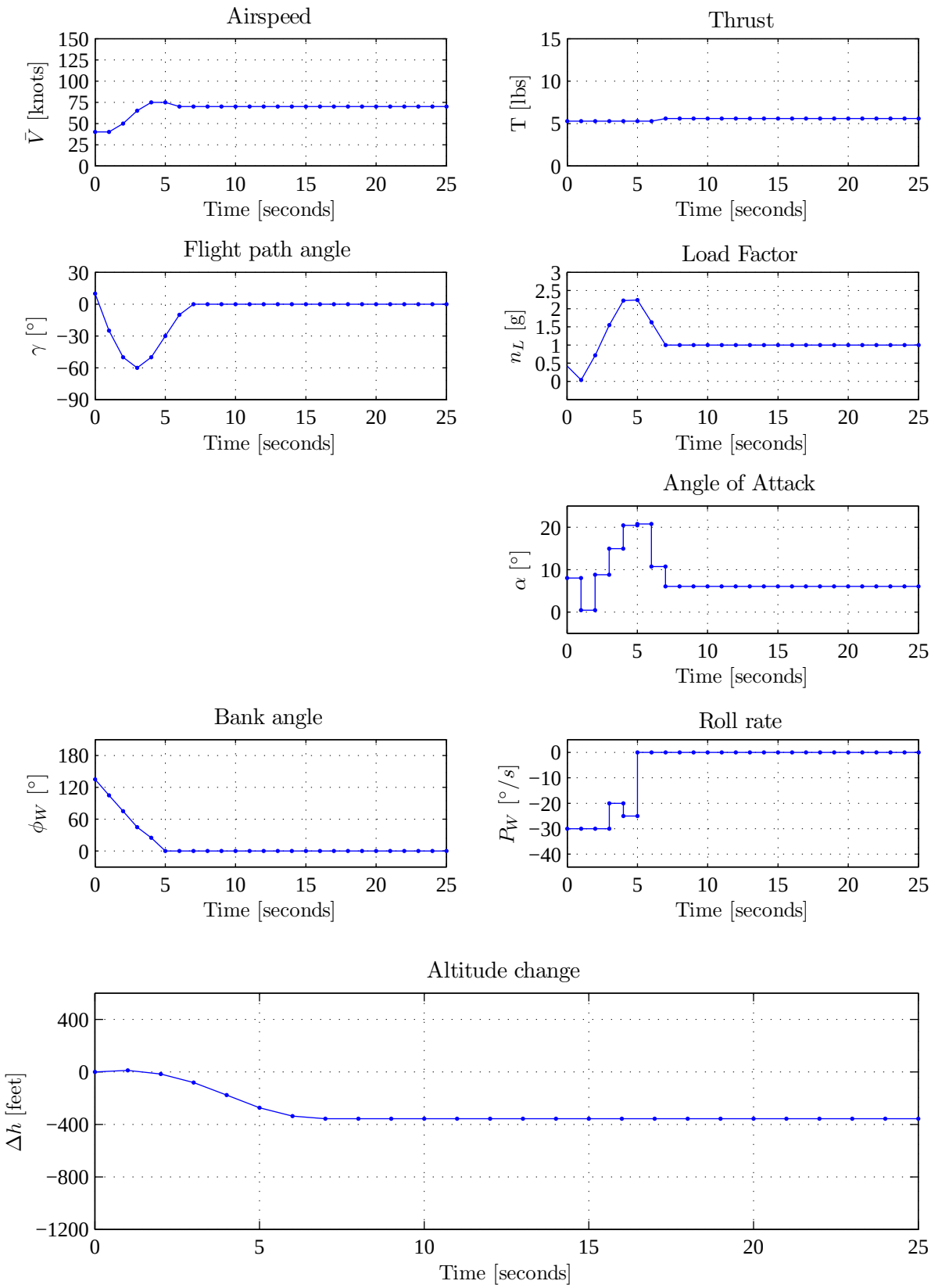


Figure 7.21: Bank angle and flight path recovery: aircraft upside-down, ascending flight path, underspeed

Flight path angle recovery, steep descent

An optimal *flight path angle recovery* from a *steep descent* is shown in figure 7.14. The aircraft was initialised with a large negative flight path angle of $\gamma = -60$ degrees, a normal airspeed of 70 knots, and a bank angle of zero (wings level). The dynamic programming control law recovers the negative flight path angle by commanding higher angles of attack to produce more lift and increase the normal load factor, causing the aircraft to pull up. The aircraft is recovered to level flight, i.e. the flight path angle is recovered to zero, within three seconds. Once the flight path angle is recovered, the angle of attack is set to its trim value to maintain level flight. During the recovery, the normal load factor does not exceed 2.5g, and remains safely within the structural integrity envelope. The airspeed increases slightly to 85 knots under the influence of gravity while the flight path angle is negative, but remains within the range of admissible final airspeeds. A zero roll rate is commanded to keep the bank angle zero throughout the recovery. The thrust command is kept constant during the recovery, and is adjusted to its trim setting once the recovery has been complete, to maintain the airspeed. The maximum altitude lost during the recovery is 238 feet.

Flight path angle recovery, steep climb

An optimal *flight path angle recovery* from a *steep climb* is shown in figure 7.15. The aircraft was initialised with a large positive flight path angle of $\gamma = 30$ degrees, a normal airspeed of 70 knots, and a bank angle of zero (wings level). The dynamic programming control law recovers from the positive flight path angle by commanding a lower angle of attack to reduce the lift and decrease the normal load factor, causing the aircraft flight path to drop. The aircraft is recovered to level flight, i.e. the flight path angle is recovered to zero, within one second. Once the flight path angle is recovered, the angle of attack is set to its trim value to maintain level flight. During the recovery, the normal load factor becomes as low as 0.34g, but remains safely within the structural integrity envelope. The airspeed remains essentially constant. A zero roll rate is commanded to keep the bank angle zero. The thrust command is kept constant during the recovery, and is only adjusted to its trim setting once the recovery has been complete, to maintain the final airspeed. The aircraft gains about 20 feet of altitude due to the initial positive flight path angle.

Bank angle recovery

An optimal *bank angle recovery* from an inverted bank angle is shown in figure 7.16. The aircraft finds itself practically upside-down, initialised with an inverted bank angle of 165 degrees, but with a level flight path ($\gamma = 0^\circ$) and a normal airspeed of 80 knots. The dynamic programming control law recovers the bank angle by commanding a roll rate of -30 degrees per second. It takes about 5 seconds to recover the bank angle to wings level. While the bank angle is inverted, the angle of attack and the associated normal load factor cannot be used to pull up the flight path angle, and can only pull the flight path angle further down. The angle of attack command is therefore kept near zero while the bank angle is greater than 90 degrees. During this time, the flight path angle drops because there is minimal lift to oppose gravity, and the airspeed increases because of the negative flight path angle and the reduced drag associated with the low angle of attack. As soon as the bank angle is recovered to below 90 degrees, i.e. when the aircraft crosses the bank angle from upside down to right-side, then higher angles of attack are commanded to start pulling the aircraft up. The maximum angle of attack is not commanded immediately, to allow the maximum roll rate command to be used for a few more samples to recover the bank angle. (Remember the roll rate command is constrained at higher angles of attack.) When the bank angle is close to wings level, the angle of attack command is increased to produce more lift and the associated load factor increases to about 2.5g. At the same time, the roll rate command

is reduced to -10 degrees per second due to the roll rate command limiting at higher angles of attack. At $t = 5$ seconds the aircraft has been returned to wings level and is pulling up as hard as the load factor limit allows, but with the airspeed still increasing due to the negative flight path angle. The flight path angle is then recovered to level flight at $t = 8.5$ seconds, and then pulled up further to a positive flight path angle of $\gamma = 20$ degrees at $t = 11$ seconds to reduce the overspeed with some gravity assistance. As the airspeed gradually reduces, the flight path angle is also gradually returned to level flight. At $t = 15$ seconds, the airspeed has been reduced to an acceptable final speed of 100 knots, and the flight path angle is returned to $\gamma = 0$ degrees. At $t = 16$ seconds, the recovery has been completed, and the angle of attack and thrust commands are set to their trim values to maintain the altitude and airspeed. During the entire recovery maneuver, the aircraft reaches a minimum flight path angle of 50 degrees, a maximum airspeed of 135 knots, and experiences a peak altitude loss of 664 feet. During the recovery, the normal load factor is kept between 0 and 2.5g, and both the airspeed and the load factor remain within the structural integrity envelope.

Underspeed recovery

An optimal *underspeed recovery* is shown in figure 7.17. The aircraft was initialised with a low airspeed of 15 knots, but with a level flight path ($\gamma = 0$ degrees) and a wings level bank angle ($\phi = 0$ degrees). The dynamic programming algorithm uses a negative flight path angle to recover the airspeed. The angle of attack is commanded to produce almost zero load factor, which causes the flight path angle to drop. An angle of attack of 5 degrees is commanded, which would normally be the trim angle of attack, but due to the low airspeed, it produces very little lift and therefore allows the flight path angle to drop under the influence of gravity. The large negative flight path causes the airspeed to increase rapidly. As the airspeed increases and approaches an acceptable airspeed, the control law starts pulling the aircraft up again to recover the flight path angle. A high angle of attack of 20 degrees is commanded resulting in load factors between 2 and 2.5 g. The airspeed is recovered to 75 knots at $t = 4$ seconds, and the flight path angle is recovered shortly afterwards at $t = 7$ seconds. At $t = 7$ seconds, the recovery has been completed, and the angle of attack and thrust commands are set to their trim values to maintain the altitude and airspeed. A zero roll rate is commanded throughout to keep the bank angle wings level. The aircraft loses about 400 feet during the underspeed recovery, and the load factor remains within the structural integrity envelope.

During the recovery, the flight path angle drops to -90 degrees, which means that the aircraft is dropping vertically downwards at one point. The dynamic programming algorithm does this because it is allowed ($\gamma = -90$ degrees is specified as an admissible state) and because it results in the minimum altitude loss. If it is considered to be too extreme, then the control law can be modified by executing the dynamic programming algorithm again, but with the minimum admissible flight path angle set to a shallower negative flight path angle. This would result in recovery trajectories that do not exceed the new lower limit on flight path angle, but at the cost of more altitude lost.

Overspeed recovery

An overspeed recovery maneuver is shown in figure 7.18. The aircraft was initialised with a high airspeed of 140 knots, but with a level flight path ($\gamma = 0$ degrees) and a wings level bank angle ($\phi = 0$ degrees). The dynamic programming algorithm uses a positive flight path angle to reduce the airspeed and recover from the overspeed condition. The angle of attack is commanded to produce higher load factors, which causes the flight path angle to be raised. Due to the high airspeed, relatively small angle of attack commands are used to produce large load factors. The positive flight path angle causes the airspeed to gradually reduce under the influence of gravity, until an admissible final airspeed is reached. At $t = 8$ seconds, the recovery

has been completed, and the angle of attack and thrust commands are set to their trim values to maintain the altitude and airspeed. A zero roll rate is commanded throughout to keep the bank angle wings level. No altitude is lost, and the aircraft actually gains about 280 feet during the underspeed recovery. The load factor remains within the structural integrity envelope.

The angle of attack command and the associated load factor "jumps around" somewhat, and is not as smooth as you would expect from a human pilot. For example, the load factor alternates between 2.2g, 1g, and 2.2g again at times $t = 0$, 1, and 2. This causes the flight path angle to increase to $\gamma = 10$ degrees, then remain constant for one sample, and then increase to $\gamma = 20$ degrees. It seems as if the dynamic programming algorithm does not care about the exact value of the flight path angle, as long as it is positive. This behaviour makes sense when one considers the cost function that is optimised by the dynamic programming algorithm. The cost function penalises altitude loss, but does not reward altitude gained. Since the overspeed recovery only gains altitude, there are a large number of state trajectories that would all result in a zero cost for the altitude loss cost function, and are therefore all equally optimal in terms of the primary cost function. All the trajectories are also equally optimal in terms of the maximum airspeed cost (all trajectories start with the maximum airspeed of 140 knots), and in terms of the bank angle cost (all trajectories maintain a zero bank angle). The dynamic programming algorithm therefore chooses the "first" optimal state trajectory that it finds and discards all subsequent state trajectories that are equally optimal. The control input sequences may therefore appear somewhat "random", as long as they result in the airspeed eventually returning to an admissible final airspeed, and as long as no altitude is lost during the overspeed recovery.

If a "smoother" control input sequence is desired that looks similar to those provided by a human pilot, then an additional cost function may be added to minimise the recovery time. This would help the dynamic programming algorithm differentiate between the large number of state trajectories that are equally optimal in terms of altitude loss, maximum airspeed reached, and bank angle recovery time, and encourage it to select the trajectory that also minimises the recovery time.

Bank angle and flight path angle recovery, steep descent

A combined flight path angle and bank angle recovery from a steep descent is shown in figure 7.19. The aircraft was initialised with a descending flight path angle of -30 degrees, an inverted bank angle of 120 degrees, and a typical airspeed of 75 knots. The dynamic programming control law first recovers the bank angle to less than 90 degrees before it starts recovering the flight path angle. The bank angle is recovered by commanding a roll rate of -30 degrees per second. While the bank angle is greater than 90 degrees, the angle of attack is commanded to make the normal load factor zero, so that the aircraft is not "pulled up" towards the ground. While the bank angle is inverted, the flight path angle drops because there is minimal lift to oppose gravity, and the airspeed increases because of the negative flight path angle and the reduced drag associated with the low angle of attack. As soon as the bank angle is recovered to below 90 degrees, the angle of attack is commanded to produce a normal load factor of 2g to start recovering the flight path angle. The bank angle and flight path angle are then recovered simultaneously. The flight path angle and bank angle are both recovered by $t = 7$ seconds, but by that time the airspeed has already increased to overspeed conditions. The dynamic programming control law then uses a positive flight path angle to reduce the airspeed to an admissible final airspeed. At $t = 15$ seconds, the recovery has been completed, and the angle of attack and thrust commands are set to their trim values to maintain the altitude and airspeed. The maximum altitude loss is 660 feet, and the net altitude loss is 400 feet. Throughout the recovery, the airspeed and load factor both remain within the structural integrity envelope.

Bank angle and flight path angle recovery, steep descent, overspeed

A combined flight path angle, bank angle and airspeed recovery from a steep descent with overspeed is shown in figure 7.20. The aircraft was initialised with a descending flight path angle of -15 degrees, a high bank angle of 70 degrees, and a high airspeed of 130 knots. The dynamic programming control law simultaneously recovers the flight path angle and the bank angle, and then uses a positive flight path angle to recover from overspeed. While the flight path angle is being recovered, the airspeed increases to 140 knots, but is then recovered to a final airspeed of 100 knots. The maximum altitude loss is 456 feet, and the net altitude loss is 178 feet.

Bank angle and flight path angle recovery, steep climb, underspeed

A combined flight path angle, bank angle and airspeed recovery from a steep climb with underspeed is shown in figure 7.21. The aircraft was initialised with a climbing flight path angle of 10 degrees, a bank angle of 130 degrees, and an underspeed of 40 knots. The dynamic programming control law uses a negative flight path angle to increase the airspeed while simultaneously recovering the bank angle. The angle of attack is commanded to produce a low load factor to allow the flight path angle to drop, while a wind-axis roll rate of -30 degrees per second is commanded to recover the positive bank angle. Once the bank angle has been reduced to below 90 degrees and when the airspeed approaches an admissible final airspeed, a high angle of attack of 20 degrees is commanded to produce a high normal load factor and recover the flight path angle. At $t = 7$ seconds, the airspeed, flight path angle and bank angle have all been recovered, and the angle of attack and thrust commands are set to their trim values. During the recovery, a minimum flight path angle of -60 degrees and maximum airspeed of 75 knots is reached, and the maximum altitude loss is 177 feet. Throughout the recovery, the airspeed and load factor both remain within the structural integrity envelope.

7.8 Limitations

When calculating the lift force L and the drag force D, the dynamic programming algorithm assumes a fixed value for the air density ρ and may produce suboptimal trajectories if there are variations in the actual air density. One way to solve this problem, is to formulate the optimal control problem to use normal load factor command n_L in the control input vector instead of angle of attack command α. The normal load factor inner-loop control law uses feedback from the normal acceleration sensor and provides robustness to uncertainty in the dynamic pressure and the aerodynamic model. Another advantage is that the normal load factor is kinematically related to the flight path angle rate, while the angle of attack is dynamically related to the flight path angle rate. The kinematic model that relates the normal load factor to the flight path angle rate contains very little uncertainty, while the dynamic model relating the angle of attack to the flight path angle rate uses an aerodynamic model and a translational dynamics model that contain significantly more uncertainty.

When generating the lookup table, the dynamic programming algorithm uses the variable $\overline{V}$ as airspeed magnitude when calculating the aerodynamic forces, but uses it as inertial (ground) speed magnitude when calculating the altitude loss. When it is implemented, the guidance law should use $\overline{V}$ as airspeed magnitude, since the flight envelope of the aircraft is specified in terms of airspeed and not in terms of ground speed. This means that the peak altitude loss will be predicted based on the airspeed, and not based on the ground speed. If the airspeed and the ground speed differ, then there will be a difference between the actual altitude lost relative to the ground and the altitude lost "relative to the air".

7.9 Conclusions

The simulation results show that the optimal attitude and flight vector recovery guidance law is able to successfully perform combinations of bank angle recovery (high and inverted bank angles), flight path angle recovery (steep descent and steep climb), and airspeed recovery (underspeed and overspeed) while adhering to the constraints of the structural integrity envelope (normal load factor and maximum airspeed). The dynamic programming guidance law minimises the altitude lost and the maximum airspeed reached during the recovery within the constraints of the aircraft dynamics and actuator limits of a large transport aircraft. The constraints on the angle of attack and normal load factor ensure that the aircraft remains within both the aerodynamic envelope and the structural integrity envelope during the attitude and flight vector recovery.

7.10 Summary of Contributions

The following contributions were made in this chapter:

- The upset recovery problem was formulated as an optimal control problem with the objective to find the optimal state trajectories and control inputs sequences that minimise the total altitude lost during the recovery maneuver.
- The aircraft flight dynamics was reduced to the point mass translational dynamics of the aircraft, with the fast rotational dynamics abstracted through time scale separation. This made the optimal control problem tractable to be solved with a dynamic programming approach.
- A novel hierarchical multi-objective cost function was introduced. A primary cost function is defined to minimise the altitude lost, a secondary cost function is defined to minimise the maximum airspeed reached, and a tertiary cost function is defined to recover the bank angle to wings level as soon as possible. The hierarchical cost function allows the altitude loss to be minimised without making any trade-offs to airspeed or bank angle. First, the altitude loss is minimised without trying to optimise overspeed avoidance or bank angle recovery. Next, if more than one solution minimise the altitude loss, then overspeed avoidance is also optimised without trying to optimise bank angle recovery. Finally, if more than one solution both minimise the altitude loss and optimise the overspeed avoidance, then the bank angle recovery is also optimised.
- The problems of underspeed and overspeed recovery were included in the optimal control formulation by including underspeed and overspeed states in the range of admissible states, but only including acceptable speeds in the set of admissible final states.
- Methods to include the aerodynamic envelope and the structural integrity envelope as constraints in the formulation of the optimal control problem were introduced.
- The advantages of using the dynamic programming approach were highlighted. The dynamic programming lookup table also indicates which initial states are unrecoverable.
- Two methods for handling the engine lag dynamics were proposed. In the first method, the engine thrust is not used a control input, and the thrust command is kept constant. In the second method, the thrust is treated as a state variable and the thrust command as a control input. The thrust variable is added as an extra dimension to the state search space, making the problem more computationally challenging for a dynamic programming approach. This second method was demonstrated by applying to longitudinal recovery cases where the bank angle is not considered.

- Several search optimisations were introduced into the dynamic programming search algorithm that reduced the computation time.
- A method for integrating the dynamic programming lookup table into a conventional flight control system was proposed. The method supplies the optimal state trajectories reconstructed from the dynamic programming lookup table as reference trajectories to the conventional middle-loop flight control laws (airspeed, flight path angle, and bank angle) and supplies the optimal control input sequences as feedforward references to the conventional inner-loop control laws (load factor, roll rate, and thrust).
- The optimal upset recovery function was verified in simulation using the NASA Generic Transport Model. The simulation results show that the optimal attitude and flight vector recovery guidance law is able to successfully perform combinations of bank angle recovery (high and inverted bank angles), flight path angle recovery (steep descent and steep climb), and airspeed recovery (underspeed and overspeed) while adhering to the constraints of the structural integrity envelope (normal load factor and maximum airspeed).

Chapter 8

Integrated Flight Envelope Recovery

In this chapter, the outer-loop Dynamic Programming guidance law from chapter 7 and the inner-loop Lyapunov control law from chapter 6 are combined into a single Integrated Flight Envelope Recovery System that is able to perform combined aerodynamic envelope recovery, and attitude, flight path angle, and airspeed recovery. The integrated system is implemented and verified in simulation on the full simulation model of the NASA Generic Transport Model. The simulation results validate the time scale separation assumption between the inner-loop recovery of the fast rotational dynamics and the outer-loop recovery of the point mass translational dynamics, and show that the integrated system is able to perform full flight envelope recovery. An example recovery scenario is used to illustrate the integrated flight envelope recovery system's ability to recover the aircraft from an upside-down stall condition (an inverted bank angle combined with a high angle of attack).

An alternative architecture is also proposed that combines the outer-loop Dynamic Programming guidance law with more conventional middle-loop and inner-loop flight control laws similar to those described in chapter 5. Conventional middle-loop controllers are used to control the airspeed, flight path angle, and bank angle to follow the optimal state trajectories, and conventional inner-loop controllers are used to control the load factor, roll rate and thrust to follow the optimal control input sequences. The middle-loop feedback control laws provide additional disturbance rejection and robustness to model uncertainty, while the feedforward commands to the inner-loop controllers compensate for the lag introduced by the middle-loop control laws.

Both integrated flight envelope recovery architectures are implemented and verified in simulation using the full NASA Generic Transport Model. The simulation results show that both proposed architectures are able to recover the aerodynamic envelope, the bank angle, the flight path angle and the airspeed with near-minimum peak altitude loss while adhering to the structural integrity envelope.

8.1 Dynamic Programming with Lyapunov Controller

An integrated flight envelope recovery system may be created by combining the Dynamic Programming guidance law and the Lyapunov controller. The Lyapunov controller is used as an inner-loop control law that performs angular rate recovery and aerodynamic envelope recovery, and also allows the angle of attack, wind-axis roll rate and sideslip angle to be commanded in an extended aerodynamic envelope. The Dynamic Programming guidance law is used as an outer-loop controller that performs the attitude envelope recovery, flight path angle recovery and airspeed recovery by providing the optimal angle of attack and wind-axis roll rate command sequences to the Lyapunov controller to recover the flight envelope while minimising the peak altitude lost during the recovery maneuver.

8.1.1 Control Architecture

The control architecture of the integrated flight envelope recovery system with Dynamic Programming outer loop and Lyapunov Controller inner loop is shown in figure 8.1.

The dynamic programming outer-loop guidance law provides the optimal angle of attack and wind-axis roll rate command sequences as angle of attack and wind-axis roll rate references for the inner-loop Lyapunov controller. The sideslip angle reference for the Lyapunov controller is set to a constant zero. The optimal angle of attack and wind-axis roll rate commands for the next sampling period are obtained from the pre-generated dynamic programming lookup table based on the airspeed, flight path angle and wind-axis bank angle at the current sampling instant. The dynamic programming guidance law uses a sampling period of one second. The angle of attack and wind-axis roll rate references are updated every sampling instant, and are held constant for the duration of the sampling period.

The optimal airspeed, flight path angle and wind-axis bank angle state trajectories are also generated by the dynamic programming guidance law, but are not used by the flight control laws. They are however recorded for information purposes, and serve as the reference trajectories that we expect the aircraft to follow.

The dynamic programming guidance law requires the flight path angle and wind-axis bank angle to be calculated from available sensor measurements so that they can be used as inputs for the dynamic programming guidance law. The flight path angle can be calculated from the ground velocity vector obtained from the onboard inertial measurement unit and the onboard GPS sensor. The wind-axis bank angle can be calculated from the roll, pitch and yaw angles obtained from the inertial sensors, and the angle of attack and sideslip angle obtained from the anemometric sensors.

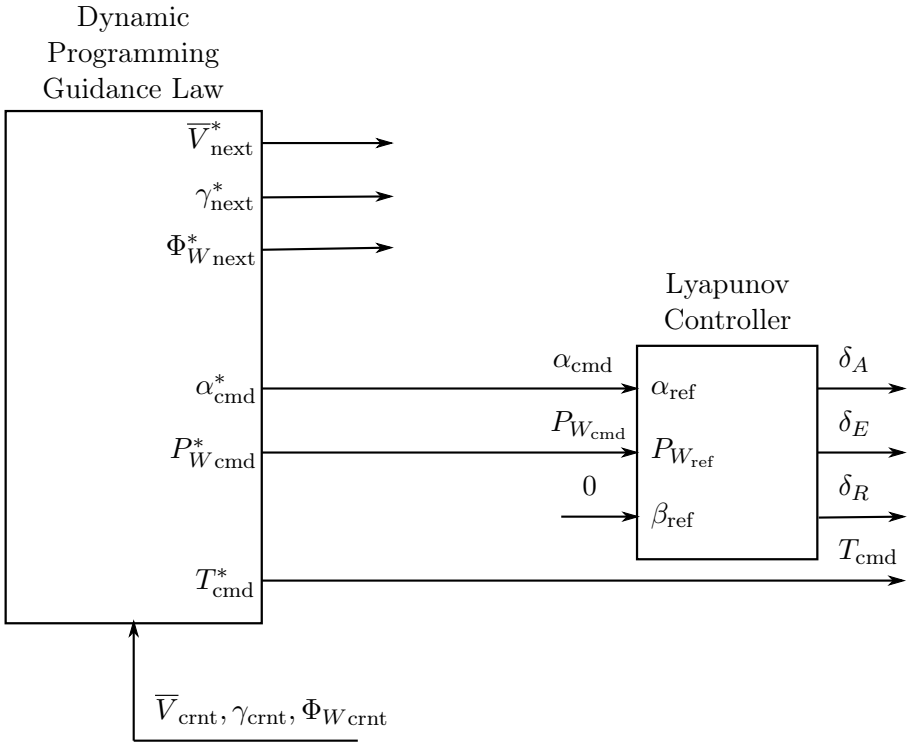


Figure 8.1: Integrated flight envelope recovery with dynamic programming outer loop and Lyapunov controller inner loop

8.1.2 Sensing the Flight Path Angle

We need to calculate the flight path angle γ from available sensor measurements so that it can be used as an input for the dynamic programming guidance law. The flight path angle can be calculated from the ground velocity vector in inertial axes as follows

$$\gamma = \arctan\left(\frac{-V_D}{\sqrt{V_N^2 + V_E^2}}\right) \tag{8.1}$$

$$\tag{8.2}$$

The ground velocity vector can be obtained from the onboard inertial measurement unit, augmented by measurements from the onboard GPS sensor. Since the flight path angle is calculated from the velocity vector of the point mass, it has the advantage of being independent of the aircraft attitude. Note that for calculating the flight path angle, we are interested in the flight path angle relative to the ground, and we therefore use the ground velocity vector instead of the airspeed vector.

8.1.3 Sensing the Wind-Axis Bank Angle

We need to calculate the wind-axis bank angle Φ_W from available sensor measurements so that it can be used as an input for the dynamic programming guidance law. We now show that the wind-axis bank angle Φ_W can be calculated given the aircraft roll Φ, pitch Θ and yaw Ψ obtained from the inertial sensors, and the angle of attack α and sideslip angle β obtained from the anemometric sensors.

We first recognise that the attitude of the wind-axis system relative to the inertial axis system $\mathbf{A}_{I\to W}$ is the product of the attitude of the body axis system relative to the inertial axis system $\mathbf{A}_{I\to B}$ and the attitude of the wind-axis system relative to the body axis system $\mathbf{A}_{B\to W}$. This can be expressed mathematically as

$$\mathbf{A}_{I\to W} = \mathbf{A}_{B\to W}\mathbf{A}_{I\to B} \tag{8.3}$$

We calculate the direction cosine matrix $\mathbf{A}_{I\to B}$ of the body axis system relative to the inertial axis system given the yaw angle Ψ, the pitch angle Θ, and the (body-axis) roll angle Φ using

$$\mathbf{A}_{I\to B} = \begin{bmatrix} \cos\Psi\cos\Theta & \sin\Psi\cos\Theta & -\sin\Theta \\ \sin\Psi\cos\Theta & \sin\Psi\sin\Theta\sin\Phi + \cos\Psi\cos\Phi & \sin\Psi\sin\Theta\cos\Phi - \cos\Psi\sin\Phi \\ \sin\Theta & \cos\Theta\sin\Phi & \cos\Theta\cos\Phi \end{bmatrix} \tag{8.4}$$

We calculate the direction cosine matrix $\mathbf{A}_{B\to W}$ of the wind axis system relative to the body axis system given the angle of attack α and the sideslip angle β using

$$\mathbf{A}_{B\to W} = \begin{bmatrix} \cos\alpha\cos\beta & -\cos\alpha\sin\beta & -\sin\alpha \\ \sin\beta & \cos\beta & 0 \\ \sin\alpha\cos\beta & -\sin\alpha\sin\beta & \cos\alpha \end{bmatrix}^{-1} \tag{8.5}$$

We then calculate the direction cosine matrix $\mathbf{A}_{I\to W}$ using matrix multiplication of the direction cosine matrices $\mathbf{A}_{I\to B}$ and $\mathbf{A}_{B\to W}$ as in equation (8.3).

$$\mathbf{A}_{I\to W} = \mathbf{A}_{B\to W}\mathbf{A}_{I\to B}$$

Finally, we recognise that the direction cosine matrix $\mathbf{A}_{I\to W}$ of the wind-axis system relative to the inertial axis system can also be expressed in terms of the flight path heading Ψ_W, the flight path angle γ, and the wind-axis bank angle Φ_W as

$$\mathbf{A}_{I\to W} = \begin{bmatrix} \cos\Psi_W\cos\gamma & \sin\Psi_W\cos\gamma & -\sin\gamma \\ \sin\Psi_W\cos\gamma & \sin\Psi_W\sin\gamma\sin\Phi_W + \cos\Psi_W\cos\Phi_W & \sin\Psi_W\sin\gamma\cos\Phi_W - \cos\Psi_W\sin\Phi_W \\ \sin\gamma & \cos\gamma\sin\Phi_W & \cos\gamma\cos\Phi_W \end{bmatrix} \tag{8.6}$$

which means that the wind-axis bank angle Φ_W can be calculated from the elements of the direction cosine matrix $\mathbf{A}_{I\rightarrow W}$ using

$$\Phi_W = arctan\frac{[\mathbf{A}_{I\rightarrow W}]_{23}}{[\mathbf{A}_{I\rightarrow W}]_{33}} \tag{8.7}$$

Note that for small angles of attack and sideslip angles, the body-axis bank angle provides a good approximation of the wind-axis bank angle and may be used as a substitute feedback variable by the wind-axis bank angle controller.

8.1.4 Simulation Results

The Dynamic Programming with Lyapunov Controller architecture was implemented and verified in simulation using NASA Generic Transport Model. The simulations were performed using the full NASA GTM simulation model made available by NASA for design and analysis purposes. The simulation model includes detailed models of the aircraft dynamics, the wide-envelope aerodynamics, the engines, the onboard sensors (including sensor bandwidth and measurement noise), and the control surface actuators (including actuator dynamics, saturation limits, and slew rate limits). Wind disturbances were added to the simulation using a simulation block that implements the MIL 8785 Dryden turbulence model.

The simulation results for an integrated flight envelope recovery from a combined aerodynamic envelope and attitude envelope upset are shown in figures 8.2 to 8.5. The aircraft initially finds itself upside-down and stalled with an inverted bank angle of -150 degrees and a high angle of attack of 50 degrees. The aircraft starts at an altitude of 800 feet, with an ascending flight path angle of 15 degrees, and a normal airspeed of 70 knots. The initial sideslip angle is zero, and the initial body angular rates are close to zero. The wind disturbance model is set to light turbulence. The integrated flight envelope recovery system is activated at time $t = 0$.

Figure 8.2 shows the time histories of the aircraft translation states, namely airspeed, flight path angle, and wind-axis bank angle. Figure 8.3 shows the time histories of the aircraft rotational states that are controlled by the Lyapunov controller, namely angle of attack, wind-axis roll rate, and sideslip angle. The same figure also shows the time history of the normal load factor, which is closely related to the angle of attack, to verify that it remains within the structural integrity envelope. Figure 8.4 shows the time history of the aircraft altitude. Figure 8.5 shows the full three-dimensional flight trajectory of the aircraft in NED position coordinates. For perspective, the projections of the aircraft's flight trajectory into the horizontal plane (North-East plane), and two vertical planes (Altitude-North and Altitude-East planes) are also shown.

The Lyapunov controller recovers the angle of attack from 50 degrees to 5 degrees within the first second, and controls the angle of attack and wind-axis roll rate to follow the optimal control input references provided by the Dynamic Programming guidance law to recover the bank angle, the flight path angle and the airspeed while minimising the peak altitude loss and remaining within the limits of the structural integrity envelope. The dynamic programming control law recovers the bank angle by commanding a roll rate of 30 degrees per second. It takes about 5 seconds to recover the bank angle to wings level. While the bank angle is greater than 90 degrees, the angle of attack is commanded in such a way that the associated load factor is kept relatively low, so that it does not pull the flight path angle further down. During this time, the flight path angle drops because there is minimal lift to oppose gravity, and the airspeed increases because of the negative flight path angle and the reduced drag associated with the low angle of attack. As soon as the bank angle is recovered to below 90 degrees, i.e. when the aircraft crosses the bank angle from upside down to rightside up, then the angle of attack is commanded to increase the normal load factor and to start pulling the aircraft up.

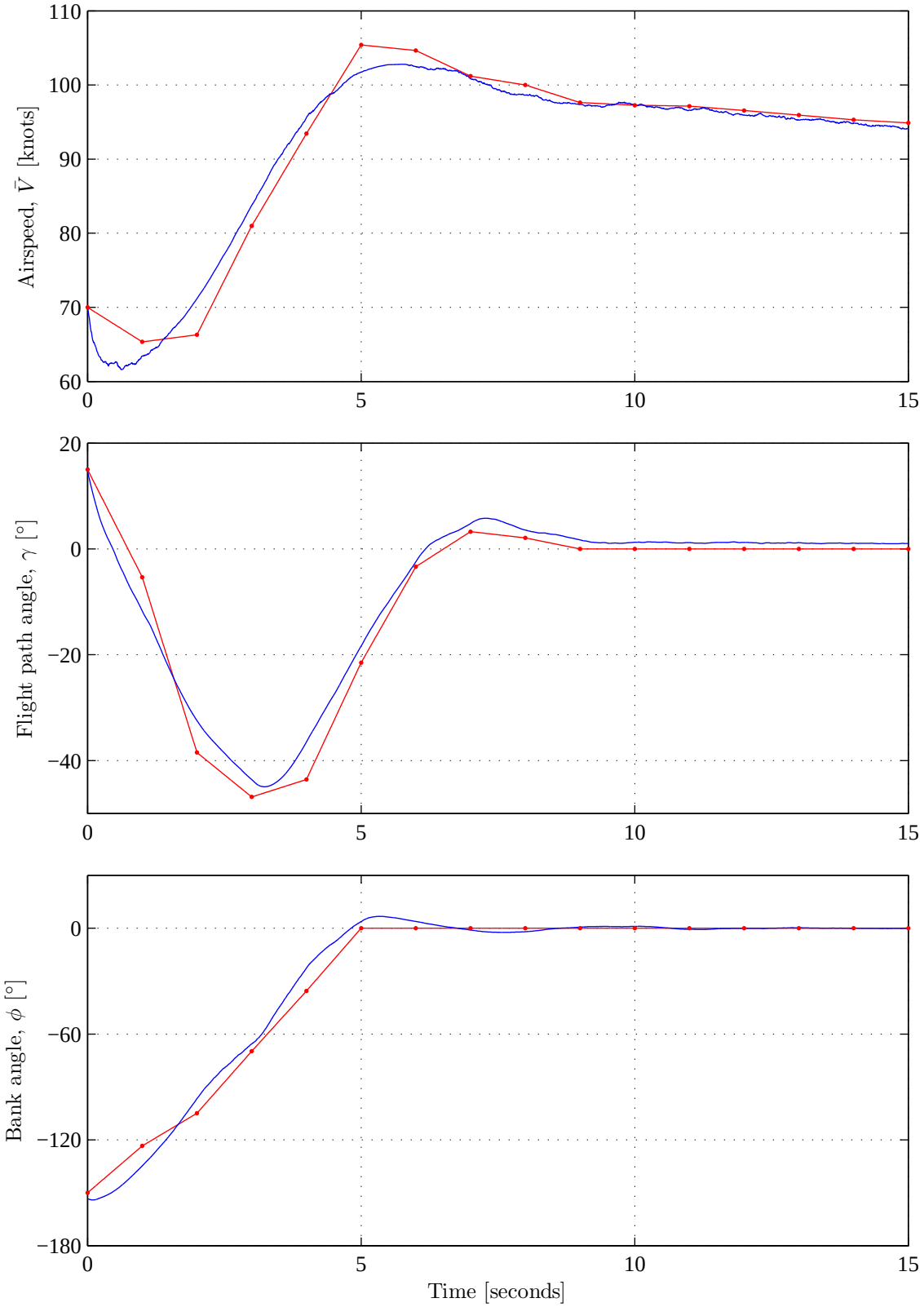


Figure 8.2: Dynamic programming with Lyapunov controller: airspeed, flight path angle, and bank angle vs. time

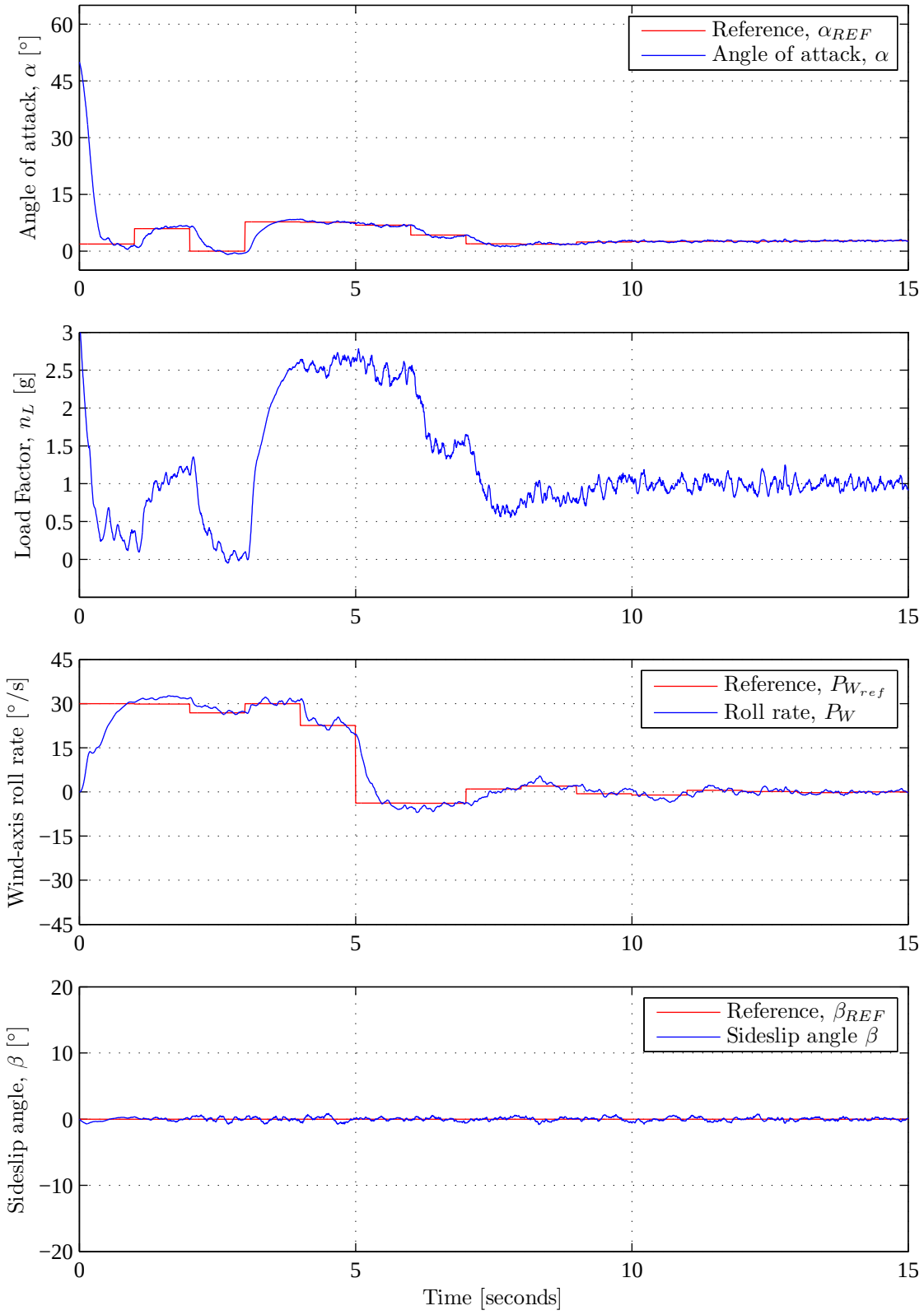


Figure 8.3: Dynamic programming with Lyapunov controller: angle of attack, load factor, wind-axis roll rate, and sideslip angle vs. time

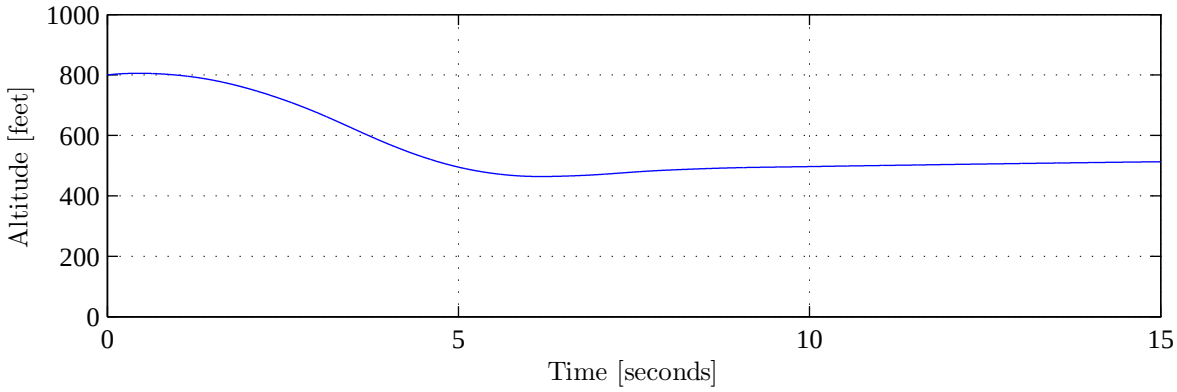


Figure 8.4: Dynamic programming with Lyapunov controller: altitude vs. time

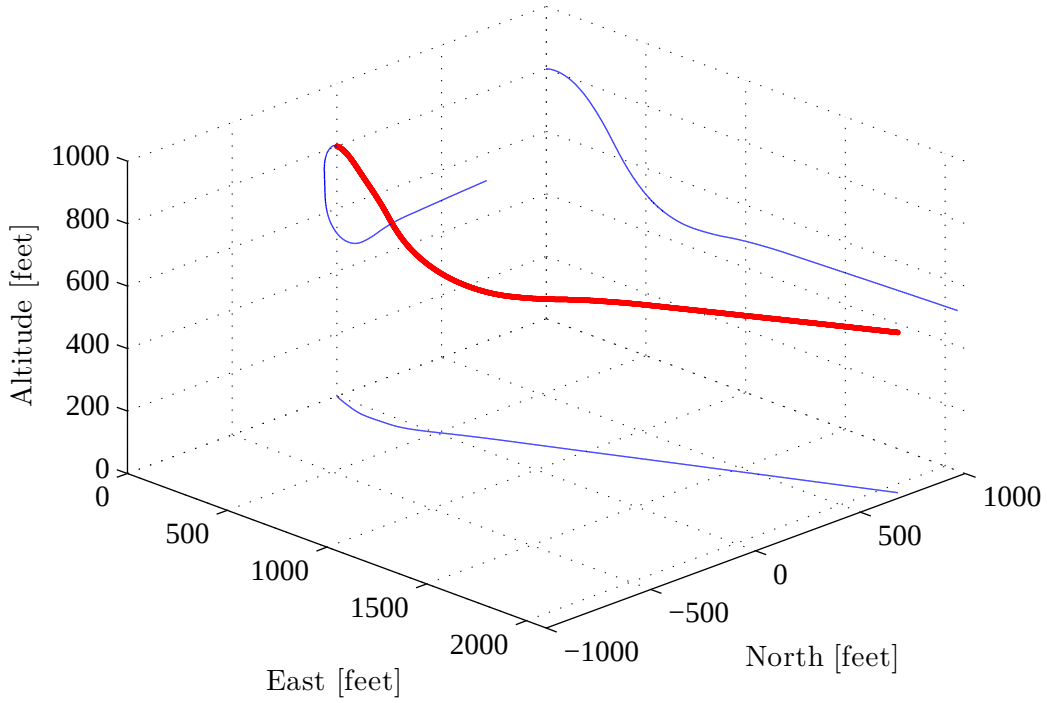


Figure 8.5: Dynamic programming with Lyapunov controller: three-dimensional flight trajectory in NED coordinates

From $t = 3$ seconds, the aircraft is pulling up with a normal load factor of 2.5g which is the maximum allowed by the structural integrity envelope. At $t = 5$ seconds the bank angle has been recovered to wings level, but the flight path angle is still negative and the airspeed is still increasing. At $t = 6.5$ seconds the flight path angle has been recovered, but by this time the aircraft has entered an overspeed condition with the airspeed exceeding 100 knots. The dynamic programming guidance law then uses a positive flight path angle to reduce the airspeed with some gravity assistance. At $t = 9$ seconds, the airspeed has been recovered to an acceptable final speed of below 100 knots, and the flight path angle is returned to level flight.

At $t = 10$ seconds, the integrated flight envelope recovery has been completed, and the angle of attack and thrust commands are set to their trim values to maintain the altitude and airspeed.

Figure 8.3 shows that the Lyapunov controller successfully controls the aircraft angle of attack and wind-axis roll rate to follow the reference commands provided by the Dynamic Programming guidance law, and also successfully regulates the sideslip angle to zero. The angle of attack and wind-axis roll rate reference commands provided by the Dynamic Programming guidance law and the zero sideslip angle reference command are plotted in red, and the actual angle of attack, wind-axis roll rate and sideslip angle are plotted in blue. The angle of attack and wind-axis roll rate transient responses both exhibit a time constant of about 0.1 seconds, which is sufficiently faster than the sampling period of the dynamic programming reference command updates. The steady-state tracking errors for both angle of attack and wind-axis roll rate are also practically zero. These simulation results validate the assumption that the time scales of the fast rigid body rotational dynamics are sufficiently shorter than the time scales of the slow point mass translational dynamics so that time scale separation can be used to perform the aerodynamic envelope recovery as an "inner loop recovery" and the attitude and flight vector recovery as an "outer loop recovery".

The time history of the normal load factor is also shown in Figure 8.3 since normal load factor is closely related angle of attack. From the plots it can be seen that the variations in the normal load factor roughly follow the variations in the angle of attack, as would be expected. The simulation results also show that the normal load factor of the aircraft is kept mostly between 0 and 2.5g as required by the structural integrity envelope. However, it is observed that the load factor does at times marginally exceed both the lower limit of 0g and the upper limit of 2.5g during the recovery. This can be explained by the external wind disturbances momentarily increasing the angle of attack and the associated normal load factor, causing the load factor to marginally exceed the load factor limits. This can be remedied by making the load factor limits a little more conservative when generating the dynamic programming lookup table, to provide some margin for the load factor variations expected to be produced by the external wind disturbances.

Figure 8.2 shows that the actual airspeed, flight path angle and wind-axis bank angle trajectories of the aircraft all follow the expected optimal reference trajectories generated by the Dynamic Programming guidance law quite closely. The optimal reference state trajectories expected by the Dynamic Programming guidance law are plotted in red, while the actual state trajectories executed by the aircraft are plotted in blue. (Remember that the Dynamic Programming with Lyapunov Controller architecture does not explicitly control the airspeed, flight path angle, and wind-axis bank angle to follow the optimal reference trajectories, but only provides angle of attack and wind-axis roll rate commands and then expects that the actual state trajectories will follow the reference state trajectories.) The largest deviation is observed in the airspeed during the first second of the recovery, and can be explained by the higher than expected drag experienced by the aircraft while the high angle of attack is still being recovered by the Lyapunov controller. It is also observed that the actual flight path angle settles to a small positive final value instead of zero as expected by the reference flight path angle, and that the airspeed does not settle to a constant final value, but slowly decreases over time. This can be explained by the fact that the optimal angle of attack commanded by the dynamic programming guidance law does not take into account the small vertical component of thrust produced because the thrust vector is not perfectly aligned with the aircraft velocity vector. The simplified point mass translational dynamics model made the simplifying assumption that the thrust vector is in the same direction as aircraft velocity vector. In reality, the thrust vector is more closely aligned with the body x-axis than the wind x-axis, and a nonzero angle of attack would therefore also produce a thrust component perpendicular to the aircraft velocity vector. The small positive flight path angle in turn causes the airspeed to slowly reduce over time due to a non-zero gravitational force component parallel to and opposing the aircraft velocity vector.

Figure 8.4 shows that the peak altitude loss during the flight envelope recovery maneuver is 366 feet, which is close to the peak altitude loss of 295 feet predicted by the Dynamic Programming lookup table for the given initial airspeed, flight path angle, and wind-axis bank angle. The small difference between the predicted and actual altitude loss can be ascribed to the fact that the prediction is based on a simplified point mass translational dynamics model that assumes that the angle of attack and wind-axis roll rate commands are followed instantaneously, while for the actual aircraft the angle of attack and wind-axis roll rate exhibit a transient with a finite speed of response.

Figure 8.5 shows the full three-dimensional flight trajectory of the aircraft during the flight envelope recovery maneuver, and illustrates the effect of the recovery maneuver on the flight path heading. The ground track projection shows that the initial flight path heading is due East, and that the flight path heading veers towards the North-East over the course of the flight envelope recovery maneuver. This is due to the fact that the aircraft is recovered from a negative bank angle and that the projection of the normal load factor into the horizontal plane is towards the left side of the ground track while the bank angle is being recovered. The aircraft flight path heading therefore changes towards the left while the bank angle is being returned to wings level.

8.2 Dynamic Programming with Conventional Control Laws

An integrated flight envelope recovery system may also be created by combining the Dynamic Programming guidance law and with conventional flight control laws, instead of the Lyapunov controller. In this architecture, the flight envelope recovery is executed in two stages, with the Lyapunov controller first recovering the angular rates and aerodynamic envelope during the first stage, and the Dynamic Programming with Conventional Flight Controllers recovering the bank angle, flight path angle, and airspeed during the second stage.

The dynamic programming algorithm is an optimal path planning algorithm, and not necessarily a robust feedback control law. The Dynamic Programming with Lyapunov Controller architecture assumes that the optimal state trajectory will be produced when the optimal control input sequences are commanded. Although it does incorporate implicit feedback control due to the fact that the optimal control inputs are determined based on the current aircraft state, it does not explicitly control the airspeed, flight path angle and wind-axis bank angle to follow the optimal state trajectories that are also provided by the dynamic programming guidance law.

For practical implementation, it is recommended that the dynamic programming guidance law be used with middle-loop feedback control laws that control the airspeed, flight path angle and the wind-axis bank angle, and inner-loop feedback control laws that control the normal load factor and the roll rate. One advantage is that the the middle-loop feedback control laws provide explicit disturbance rejection and robustness to model uncertainty. Another advantage is that the dynamic programming guidance law can then be used with existing fly-by-wire flight control laws with all of their flight envelope protection functions already included.

8.2.1 Control Architecture

The block diagram of the integrated flight envelope recovery system with dynamic programming outer guidance loop and conventional middle-loop and inner-loop flight control laws is shown in figure 8.6.

The dynamic programming guidance law generates the optimal state trajectories for the airspeed, the flight path angle and the wind-axis bank angle, and also generates the optimal control sequences for the angle of attack command, the wind-axis roll rate command, and the thrust command. (The angle of attack command is then translated into a normal load factor

command so that it can be fed forward to the inner-loop load factor controller.) The optimal airspeed, optimal flight path angle and optimal wind-axis bank angle state trajectories are provided as references to the airspeed controller, the flight path angle controller and the wind-axis bank angle controller, and the optimal command sequences for the normal load factor, wind-axis roll rate and thrust are provided as feedforward commands to the inner-loop normal load factor controller, roll rate controller and engine controller. The middle-loop feedback control laws will then provided additional disturbance rejection and robustness to model uncertainty, while the feedforward commands to the inner-loop controllers will compensate for the lag introduced by the middle-loop control laws.

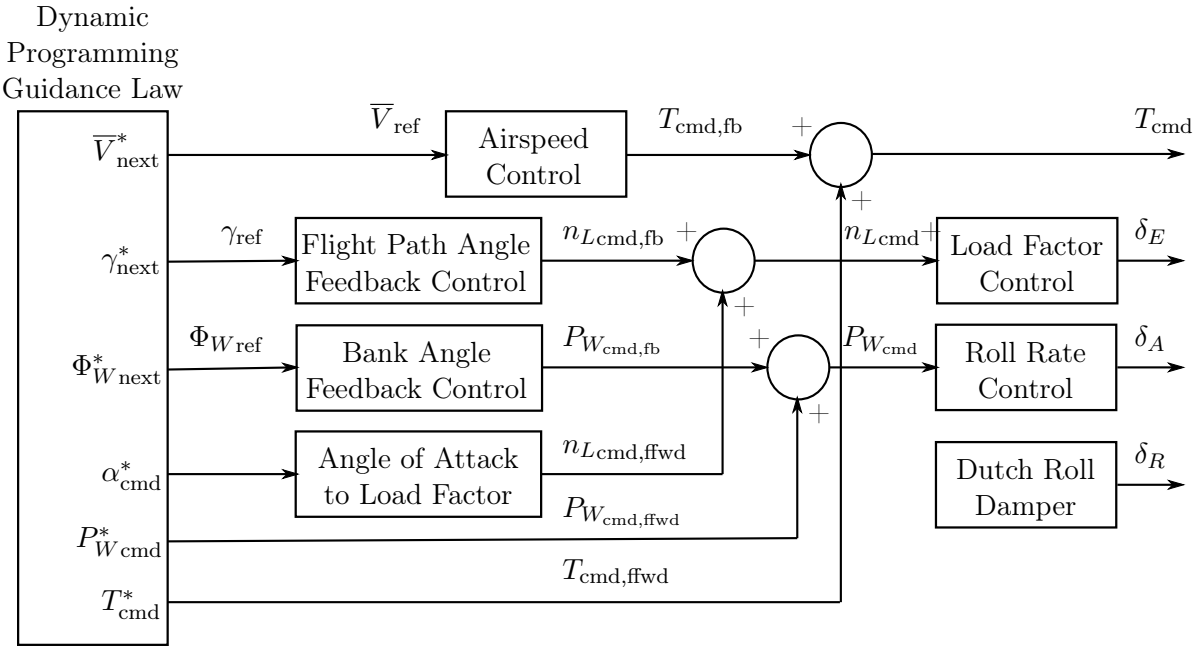


Figure 8.6: Integrated flight envelope recovery architecture with dynamic programming outer loop and conventional flight control middle and inner loops

8.2.2 Simulation Results

The Dynamic Programming with Conventional Control Laws architecture was also implemented and verified in simulation using NASA Generic Transport Model. The simulation results for an integrated flight envelope recovery from an inverted bank angle are shown in figures 8.7 to 8.10. The aircraft initially finds itself upside-down with an inverted bank angle of -150 degrees. The aircraft starts at an altitude of 800 feet, with a level flight path angle of 0 degrees, and a normal airspeed of 80 knots. The initial sideslip angle is zero, and the initial body angular rates are close to zero. The wind disturbance model is set to light turbulence. The integrated flight envelope recovery system is activated at time $t = 0$.

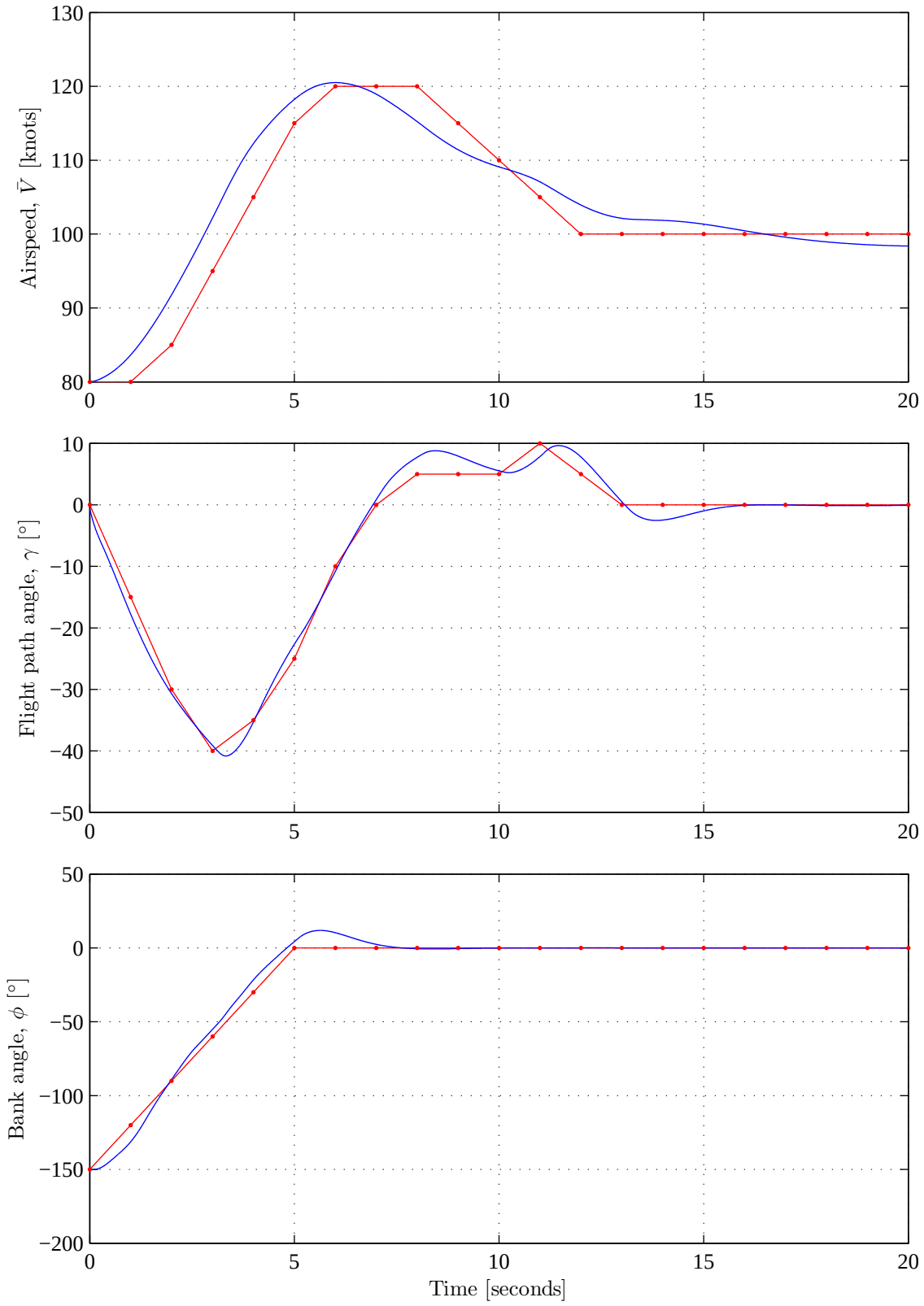


Figure 8.7: Dynamic programming with conventional control laws: airspeed, flight path angle, and bank angle vs. time

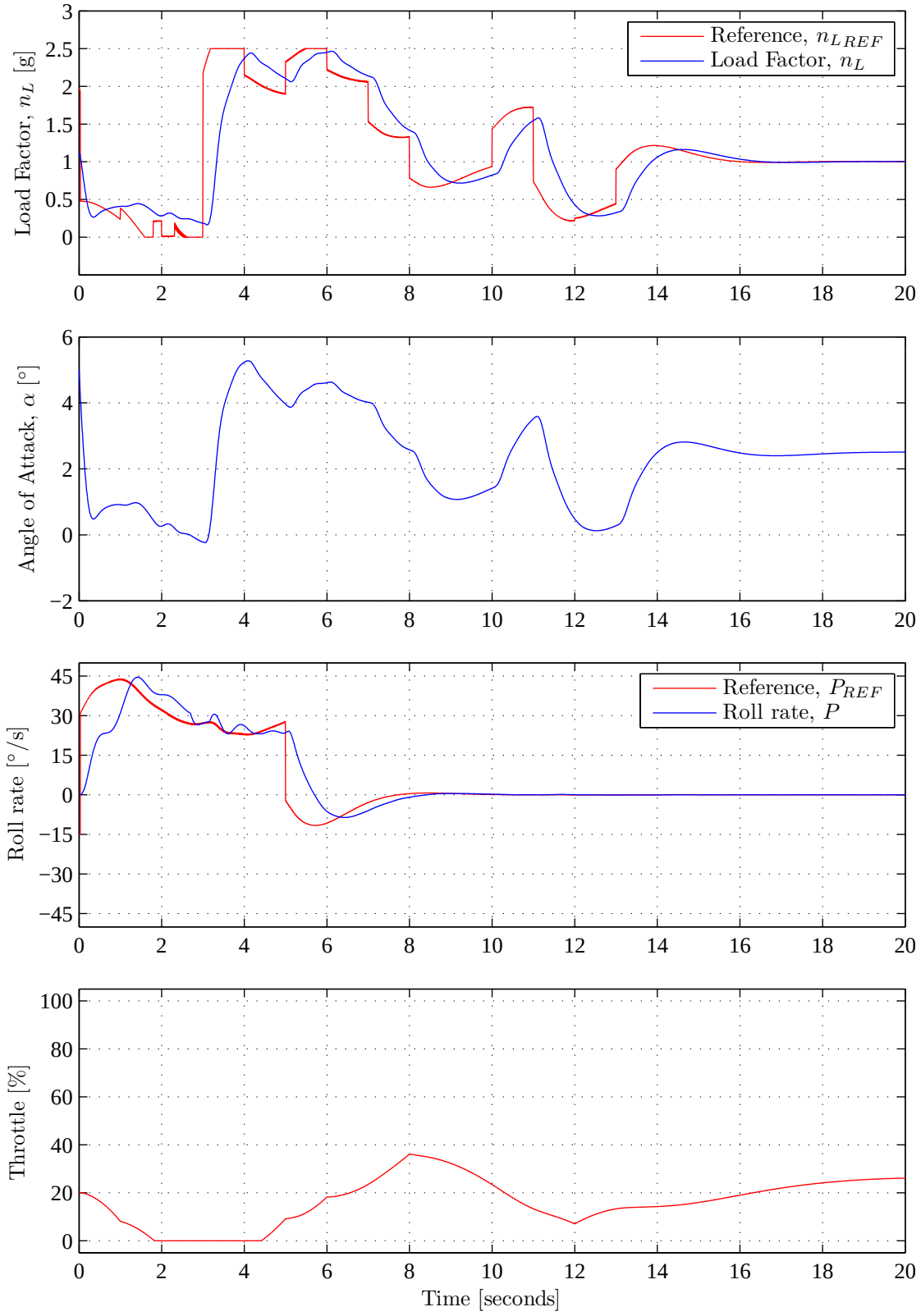


Figure 8.8: Dynamic programming with conventional control laws: load factor, angle of attack, roll rate and throttle vs. time

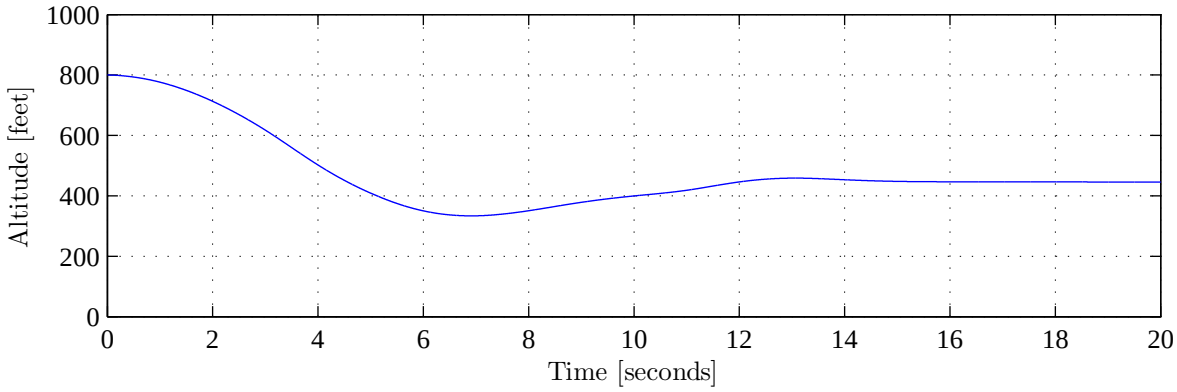


Figure 8.9: Dynamic programming with conventional control laws: altitude vs. time

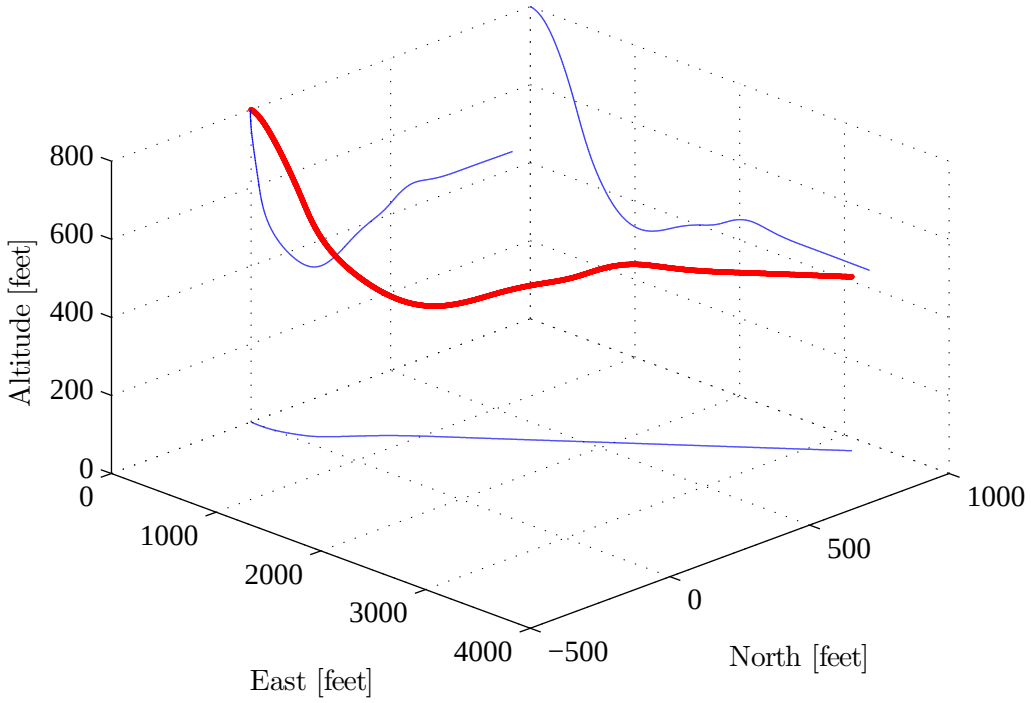


Figure 8.10: Dynamic programming with conventional control laws: three-dimensional flight trajectory in NED coordinates

Figure 8.7 shows the time histories of the aircraft airspeed, flight path angle, and wind-axis bank angle. Figure 8.8 shows the time histories of the normal load factor, roll rate, and throttle. The same figure also shows the time history of the angle of attack, which is closely related to the normal load factor, to verify that it remains within the aerodynamic envelope. Figure 8.9 shows the time history of the aircraft altitude. Figure 8.10 shows the full three-dimensional flight trajectory of the aircraft in NED position coordinates. For perspective, the projections of the aircraft's flight trajectory into the horizontal plane (North-East plane), and two vertical planes (Altitude-North and Altitude-East planes) are also shown.

The dynamic programming control law recovers the bank angle by commanding a roll rate of 30 degrees per second. It takes about 5 seconds to recover the bank angle to wings level. While the bank angle is greater than 90 degrees, the normal load factor is kept close to zero, so that it does not pull the flight path angle further down. While the bank angle is being recovered, the flight path angle drops because there is minimal lift to oppose gravity, and the airspeed increases because of the negative flight path angle and the reduced drag associated with the low angle of attack. As soon as the bank angle is recovered to below 90 degrees, i.e. when the aircraft crosses the bank angle from upside down to rightside up, then the normal load factor is increased to start pulling the aircraft up. From $t = 3$ seconds, the aircraft is pulling up with a normal load factor close to 2.5g which is the maximum allowed by the structural integrity envelope. At $t = 5$ seconds the bank angle has been recovered to wings level, but the flight path angle is still negative and the airspeed is still increasing. At $t = 7$ seconds the flight path angle has been recovered, but by this time the aircraft has entered an overspeed condition with the airspeed exceeding 100 knots. The dynamic programming guidance law then uses a positive flight path angle to reduce the airspeed with some gravity assistance. At $t = 16$ seconds, the airspeed has been recovered to an acceptable final speed of below 100 knots. Once the recovery has been completed, the airspeed reference is kept at 100 knots, the flight path angle is set to 0 degrees, and the bank angle reference is set to 0 degrees to maintain straight and level flight.

Figure 8.7 shows that the middle-loop controllers successfully control the actual airspeed, flight path angle and wind-axis bank angle trajectories of the aircraft to follow the optimal reference state trajectories provided by the Dynamic Programming guidance law. The optimal reference state trajectories provided by the Dynamic Programming guidance law are plotted in red, while the actual state trajectories executed by the aircraft are plotted in blue.

Figure 8.8 shows that the inner-loop controllers successfully control the normal load factor and roll rate to follow the optimal control input sequences provided by the Dynamic Programming guidance law. The normal load factor and roll rate references provided by the Dynamic Programming guidance law are plotted in red, and the actual normal load factor and roll rate of the aircraft are plotted in blue. The load factor plot shows that the normal load factor remains between 0 and 2.5g and therefore adheres to the structural integrity envelope. It is also observed that the thrust command is allowed to vary. Although the feedforward thrust command provided by the Dynamic Programming guidance law is kept constant, the middle-loop airspeed controller commands the thrust to control the airspeed to follow the optimal reference airspeed provided by the Dynamic Programming guidance law.

The time history of the angle of attack is also shown in Figure 8.8 since angle of attack is closely related to normal load factor. From the plots it can be seen that the variations in the angle of attack roughly follow the variations in the normal load factor, as would be expected. The simulation results show that the angle of attack remains between 0 and 6 degrees which is well within the aerodynamic envelope.

Figure 8.9 shows that the peak altitude loss during the flight envelope recovery maneuver is 429 feet, which is close to the peak altitude loss of 466 feet predicted by the Dynamic Programming lookup table for the given initial airspeed, flight path angle, and wind-axis bank angle. The small difference between the predicted and actual altitude loss can be ascribed to the fact that the actual aircraft state trajectory does not follow the optimal reference state trajectory with zero error.

Figure 8.10 shows the full three-dimensional flight trajectory of the aircraft during the flight envelope recovery maneuver, and illustrates the effect of the recovery maneuver on the flight path heading. The ground track projection shows that the initial flight path heading is due East, and that the flight path heading veers towards the North-East over the course of the flight envelope recovery maneuver. This is due to the fact that the aircraft is recovered from a negative bank angle and that the projection of the normal load factor into the horizontal plane is towards the left side of the ground track while the bank angle is being recovered. The aircraft

flight path heading therefore changes towards the left while the bank angle is being returned to wings level.

8.3 Conclusions

The simulation results for the Dynamic Programming with Lyapunov Controller architecture confirm that the Lyapunov controller can be successfully integrated with the dynamic programming guidance law, and that the resulting Integrated Flight Envelope Recovery System is able to successfully perform a combined aerodynamic envelope, attitude envelope, and flight vector recovery with near-minimal altitude loss while adhering to the structural integrity envelope of the aircraft. The simulation results validate the assumption that the time scale separation between the fast rigid body rotational dynamics and the slow point mass translational dynamics can be used to perform the aerodynamic envelope recovery as an inner loop recovery and the attitude and flight vector recovery as an outer loop recovery.

The simulation results for the Dynamic Programming with Conventional Control Laws architecture show that the dynamic programming guidance law can also be used with conventional flight control laws, and that the resulting system is also able to successfully perform attitude and flight vector recovery with near-minimum altitude loss while adhering to the structural integrity envelope of the aircraft.

8.4 Summary of Contributions

The following contributions were made in this chapter:

- The outer-loop Dynamic Programming guidance law from chapter 7 and the inner-loop Lyapunov control law from chapter 6 were combined into a single Integrated Flight Envelope Recovery System that is able to perform combined aerodynamic envelope recovery, attitude envelope recovery and airspeed recovery.
- The Dynamic Programming with Lyapunov Controller architecture was implemented and verified in simulation on the full simulation model of the NASA Generic Transport Model, with representative sensor models, actuator models, and wind disturbances.
- The simulation results validated the assumption that the time scales of the fast rigid body rotational dynamics are sufficiently shorter than the time scales of the slow point mass translational dynamics so that time scale separation can be used to perform the aerodynamic envelope recovery as an "inner loop recovery" and the attitude and flight vector recovery as an "outer loop recovery".
- An example recovery scenario was used to illustrate that the Dynamic Programming with Lyapunov Controller architecture is able to recover the aircraft from an upside-down stall condition (an inverted bank angle combined with a high angle of attack).
- An alternative architecture was also proposed that integrates the outer-loop Dynamic Programming guidance law with more conventional middle-loop and inner-loop flight control laws. Conventional middle-loop controllers are used to control the airspeed, flight path angle, and bank angle to follow the optimal state trajectories, and conventional inner-loop controllers are used to control the load factor, roll rate and thrust to follow the optimal control input sequences. The middle-loop feedback control laws provide additional disturbance rejection and robustness to model uncertainty, while the feedforward commands to the inner-loop controllers compensate for the lag introduced by the middle-loop control laws.

- The Dynamic Programming with Conventional Control Laws architecture was implemented and verified in simulation on the full simulation model of the NASA Generic Transport Model, with representative sensor models, actuator models, and wind disturbances.
- The simulation results for the Dynamic Programming with Conventional Control Laws architecture show that the dynamic programming guidance law can also be used with conventional flight control laws, and that the resulting system is also able to successfully perform attitude and flight vector recovery with near-minimum altitude loss while adhering to the structural integrity envelope of the aircraft.

Chapter 9

Conclusions and Recommendations

9.1 Summary

In this thesis, a general flight envelope recovery strategy for large transport aircraft was proposed and novel guidance and flight control laws were designed and evaluated to recover the aircraft state to the normal flight envelope from out-of-envelope conditions.

The NASA Generic Transport Model (GTM), a wide-envelope aircraft model for large transport aicraft, was used as the basis for the research. The NASA GTM is a 5.5% dynamically scaled unmanned aerial vehicle model of a generic large transport aircraft that was developed by NASA for experimental flight tests outside the normal flight envelope that are considered too risky for full-scale manned aircraft testing. Wide-envelope aerodynamic models were obtained for the GTM from wind tunnel tests conducted at the NASA Langley Research Center. Steady data were obtained for different configurations and an extensive range of angles of attack, sideslip angles, and control surface deflections, while angular rate effects were modeled by blending data sets obtained from oscillatory motion rigs and steady motion rigs. From these aerodynamic models, a full non-linear wide-envelope flight dynamics simulation model of the GTM was created and made available by NASA for research use.

A two-tier flight envelope recovery strategy was proposed that first performs high angular rate recovery and aerodynamic envelope recovery, and then performs the attitude recovery, flight path angle recovery, and airspeed recovery. The two-tier approach to flight envelope recovery was motivated by the time scale separation between the fast rigid body rotational dynamics and the slow point mass translational dynamics. We argued that the angular rate and aerodynamic envelope recovery is performed through controlling the fast rotational dynamics, and that the attitude, flight path angle and airspeed recovery are performed through controlling the point mass dynamics. We also argued that the angular rate and aerodynamic envelope recovery should be prioritised, so that the conventional flight control laws and envelope protection functions are available again to perform the attitude, flight path angle and airspeed vector recovery.

Two approaches to flight envelope recovery were proposed. The first approach uses a passive method to recover the angular rates and the aerodynamic envelope using the natural stability of the aircraft, and then uses a state machine and the conventional flight control laws, with all their protection functions, to recover the attitude, flight path angle and airspeed. The second approach uses an active method to recover the aerodynamic envelope using a Lyapunov-based inner-loop controller, and uses an outer-loop controller based on optimal control theory to recover the attitude, flight path angle, and airspeed while minimising the altitude loss.

9.1.1 Passive Angular Rate and Aerodynamic Envelope Recovery

For the passive angular rate and aerodynamic envelope recovery, it was postulated that the natural angular rate damping of the aircraft could be used to recover from high angular rates,

and that the natural stability of the aircraft could be used to recover the angle of attack and sideslip angle. The hypothesis was that the aircraft will be attracted to a stable equilibrium with low angular rates and a desirable angle of attack and sideslip angle inside the aerodynamic envelope. However, the danger was that the aircraft may have multiple stable equilibria, and may also naturally be attracted to undesirable angles of attack and sideslip angles outside the aerodynamic envelope, depending on the initial conditions.

A bifurcation analysis was performed to find the desirable stable equilibria of the aircraft inside the aerodynamic envelope, and to check for undesirable stable equilibria outside the envelope. For the control surfaces set to their neutral deflections and with the throttle set to idle, only one desirable stable equilibrium is found inside the aerodynamic envelope, and no undesirable stable equilibria are found outside the envelope. For excessive nose-up elevator deflections, two undesirable stable equilibrium branches are found corresponding to stable stall and spin conditions outside the aerodynamic envelope.

The regions of attraction of the stable equilibria, both inside and outside the aerodynamic envelope, were explored with nonlinear simulations by generating state trajectories and by performing Monte Carlo simulations. For the control surfaces set to their neutral deflections, the simulations showed that all trajectories converge to the stable equilibrium inside the aerodynamic envelope and that no trajectories diverge to infinity. For excessive elevator deflections, the simulations showed that all trajectories converge to either of the two stable spin equilibria outside the aerodynamic envelope and that no trajectories diverge to infinity.

The bifurcation analysis and nonlinear simulations therefore verify that passive angular rate and aerodynamic envelope recovery can be performed by returning the control surfaces to their neutral deflections and setting the throttle to idle.

9.1.2 State Machine Based Attitude and Flight Vector Recovery

A state machine based approach for flight envelope recovery was then presented that recovers the flight envelope in stages and uses the passive aerodynamic envelope recovery as the first stage of the recovery. Passive angular rate recovery and aerodynamic envelope recovery is performed by using the natural rate damping and aerodynamic stability of the aircraft. Once the aerodynamic envelope is recovered, the control surfaces produce forces and moments that behave linearly again, and the normal flight control laws and protection functions are available to perform the attitude, flight path angle and airspeed recovery. Attitude recovery is performed by using roll rate control and bank angle control to return the aircraft to wings level, and by using normal load factor control and flight path angle control to return the flight path angle to level flight. An angle of attack protection controller is used to prevent the aircraft from exiting the aerodynamic envelope again, and the normal load factor control is used to constrain the load factor and prevent structural failure and injury to the pilot and passengers. Once the bank angle and flight path angle have been recovered, an ascending flight path angle is commanded and gravity is used to rapidly recover from overspeed. Once overspeed is recovered, a level flight path angle is commanded and the airspeed controller automatically increase the engine throttle to maintain the airspeed.

The state machine based flight envelope recovery was implemented and verified on the NASA GTM. The simulation results show that the state machine and conventional flight control laws are able to recover the attitude, flight path angle, and airspeed of the aircraft, and that the normal load factor remains within the structural integrity envelope during the entire upset recovery process. However, it should be noted that the state machine based recovery is not optimised to minimise the peak altitude lost during the recovery.

9.1.3 Active Angular Rate and Aerodynamic Envelope Recovery

An active aerodynamic envelope recovery method was proposed that employs a Lyapunov-based inner-loop controller to actively recover the angular rates, the angle of attack, and the sideslip angle from out-of-envelope conditions using the control surfaces of the aircraft. The problem was reduced to controlling the rotational dynamics of the aircraft by exploiting the time scale separation between the rigid body rotational dynamics and the point mass translational dynamics of the aircraft. A novel Lyapunov function was proposed that represents the state of the fast rotational dynamics of the aircraft as an energy-like function. Lyapunov-based control laws were then derived to control the time derivative of the energy to always be negative and proportional to the energy, thus providing exponential Lyapunov stability. The resulting Lyapunov-based inner-loop controller not only actively recovers the angular rates and the aerodynamic envelope, but allows the angle of attack, sideslip angle, and wind-axis roll rate to be actively controlled in an extended aerodynamic envelope.

The actuator deflections are solved using a multi-variable secant method that takes actuator saturation into account. When the actuators do not saturate, the Lyapunov controller provides exponential stability. When the actuators saturate, the Lyapunov controller still attempts to provide at least asymptotic stability. A novel method for sensing the rotational motion of the wind axis system relative to the inertial axis system using the available onboard inertial sensors was introduced, to improve the body axis tracking of the wind axis. The Lyapunov-based controller implicitly provides gain scheduling for different airspeeds and altitudes, and also implicitly handles inertial coupling.

Extensive simulation tests were performed on the NASA Generic Tranport Model to verify the Lyapunov-based controller's ability to perform high angular rate recovery and aerodynamic envelope recovery. The simulation results show that the Lyapunov controller is able to recover the aircraft from initial states with angles of attack from -5 to +60 degrees, sideslip angles from -45 to +45 degrees, and wind-axis angular rates from -180 to +180 degrees per second.

Extensive simulation tests were also performed to verify the Lyapunov-based controller's ability to perform command tracking for the angle of attack, wind-axis roll rate, and sideslip angle in an extended aerodynamic envelope. The simulation results show that the Lyapunov controller is able to simultaneously and accurately control the angle of attack to track commands from 0 to 21 degrees, the wind-axis roll rate to track commands from -90 to +90 degrees per second, and the sideslip angle to track commands from -8 to +8 degrees.

9.1.4 Optimal Attitude and Flight Vector Recovery

An optimal attitude and flight vector recovery guidance law was proposed and designed that simultaneously recovers the bank angle, flight path angle, and airspeed of the aircraft from upset conditions while minimising the total altitude lost during the recovery maneuver.

The recovery problem was formulated as an optimal control problem and then solved using a dynamic programming algorithm to find the optimal state trajectories and the optimal sequence of control inputs. The aircraft dynamics was simplified to the reduced-order model describing the slower point mass translational dynamics of the aircraft, while the fast rotational dynamics are abstracted through time scale separation. The point mass translational dynamics incorporates the nonlinear wide-envelope aerodynamic model for the aerodynamic lift, drag and side force. The reduced-order model of the aircraft dynamics made the optimal control problem tractable to be solved with dynamic programming.

A novel hierarchical multi-objective cost function was introduced to represent the primary objective to minimise the total altitude lost, the secondary objective to minimise the maximum airspeed reached, and the tertiary objective to return the bank angle to zero as quickly as possible. The hierarchical cost function allows the primary objective to be prioritised without making any compromises to the secondary or tertiary objectives. (In contrast, a single multi-

objective cost function would typically lead to trade-offs between the primary, secondary, and tertiary objectives to minimise the total cost based on some weighting scheme.)

The airspeed, flight path angle, and bank angle limits were specified by defining the sets of admissible states. The control input limits were specified by defining sets of admissible inputs. The requirement to recover the aircraft to straight and level flight (zero flight path angle, zero bank angle) and to within an acceptable airspeed range were translated into a set of admissible final states. The problems of underspeed and overspeed recovery were included in the problem formulation by including underspeed and overspeed states in the range of admissible states, but only including acceptable speeds in the set of admissible final states. The structural integrity envelope was also included in the problem formulation. The load factor constraint was incorporated in a novel way by limiting the set of admissible angle of attack control inputs as a function of the current airspeed state. The potential aerodynamic limits of the inner-loop Lyapunov controller were also taken into account by limiting the maximum admissible wind-axis roll rate command as a function of angle of attack command. Two approaches for dealing with significant engine lag dynamics are proposed. In the first approach, the engine thrust is not used as a control input, and the thrust command is kept constant. In the second approach, the thrust is treated as a state variable and the thrust command as a control input. The thrust variable increases the dimension of the state vector by one, and makes the problem more computationally challenging for a dynamic programming approach.

The dynamic programming approach has several advantages. Firstly, the dynamic programming algorithm searches a solution grid of the entire admissible state space, and effectively produces a lookup table of the optimal state trajectories and optimal control sequences for recovery from all possible initial states. Secondly, the lookup table explicitly stores the altitude that will be lost during optimal recovery for all states. Given the state of the aircraft, the lookup table therefore predicts how much altitude would be lost if the recovery procedure is activated immediately. Thirdly, although the dynamic programming search algorithm is computationally intensive, it produces a lookup table that is computationally light in comparison. This allows the computationally heavy "design" of the lookup table to be performed offline, and the pre-computed lookup table to be used in a computationally light online implementation.

The optimal attitude and flight vector recovery guidance law was verified in simulation using the NASA Generic Transport Model. The simulation results show that the guidance law is able to successfully perform combinations of flight path angle recovery (steep descent, steep climb), bank angle recovery (high and inverted bank angles), underspeed recovery, and overspeed recovery while obeying the constraints of the structural integrity envelope. Maps of the recoverable and unrecoverable initial states were presented, as well as a statistical distributions of the peak altitude loss and peak overspeed over all recoverable trajectories.

9.1.5 Integrated Flight Envelope Recovery

Finally, the inner-loop Lyapunov control law and the outer-loop Optimal Recovery Guidance Law were combined into a single Integrated Flight Envelope Recovery System and verified in simulation on the NASA Generic Transport Model. The simulation results show that the integrated flight envelope recovery system is able to successfully recover the aircraft from an upside-down stall condition (an inverted bank angle combined with a high angle of attack) while minimising the peak altitude loss and peak overspeed. The simulation results validate the assumption that the time scale separation between the fast rigid body rotational dynamics and the slow point mass translational dynamics can be used to perform the angular rate and aerodynamic envelope recovery as an "inner loop recovery" and the attitude and flight vector recovery as an "outer loop recovery".

9.2 Future Work

We identify and briefly discuss the following promising avenues for future work:

1. The bifurcation analysis should also be performed for other aircraft configurations, such as the T-tail configuration, where there is a higher likelihood of finding stable equilibrium branches that cannot be escaped by simply returning the elevators to their trim position. The NASA Generic Transport model is not susceptible to deep stall where the aircraft stalls at high angles of attack and nose-down pitching moment cannot be generated with the application of full nose-down elevator deflection. If the NASA GTM did suffer from deep stall, then the bifurcation diagram would show more than one stable equilibrium branch for elevator deflections near zero, one desirable stable equilibrium branch inside the aerodynamic envelope, and another undesirable stable equilibria outside the aerodynamic envelope.

2. An optimal control formulation and trajectory planning approach can be applied to the problem of deep stall recovery. It is likely that the active aerodynamic envelope recovery using the Lyapunov-based controller would not be able to recover the aeodynamic envelope from deep stall due to the ineffectiveness of the elevators to produce pitching moment. A possible strategy to escape from deep stall would be to yaw the horizontal tailplane out of the wake using the rudder to regain elevator effectiveness, and then to use the elevators to recover from the stall. The aerodynamic envelope recovery could be formulated as an optimal control problem and a reduced-order model of the aircraft's fast rotational dynamics could be used to make the problem tractable to be solved with a trajectory planning algorithm. Trajectory planning techniques such as dynamic programming, sequential quadratic programming, and sampling-based path planning could be investigated.

3. The optimal attitude and flight vector recovery problem could be solved using a higher-dimensional model of the point mass translational dynamics that includes states for the engine lag dynamics, the angle of attack dynamics, and the roll rate dynamics. This higher-dimensional model would likely not be tractable to be solved using dynamic programming due to the "curse of dimensionality", but other approaches to solving the optimal control problem, such as the calculus of variations approach, the sequential quadratic programming approach, and the sampling based-path planning approach, can be investigated.

4. The integration of the outer-loop optimal recovery guidance law with conventional middle-loop flight path angle and bank angle flight control laws and inner-loop normal load factor and roll rate control laws can be investigated in more depth. The middle-loop feedback control laws would provided improved disturbance rejection and robustness to model uncertainty, while the feedforward commands to the inner-loop controllers would compensate for the lag introduced by the middle-loop control laws.

5. The optimal control problem could be reformulated to use normal load factor as an input instead of angle of attack. The relationship between the normal load factor and the flight path angle rate is a kinematic relationship that contains very little uncertainty, whereas the relationship between the angle of attack and the flight path angle rate is based on the wide-envelope aerodynamic model which inherently contains much more uncertainty. Furthermore, the optimal recovery guidance law would then give the optimal control input in terms of a load factor reference which is directly compatible with the conventional control laws used on modern fly-by-wire commercial airliners.

Bibliography

[1] James Wilborn and John Foster. Defining commercial transport loss-of-control: A quantitative approach. In *Guidance, Navigation, and Control and Co-located Conferences*. American Institute of Aeronautics and Astronautics, August 2004.

[2] Austin Murch and John Foster. Recent nasa research on aerodynamic modeling of post-stall and spin dynamics of large transport airplanes. In *Aerospace Sciences Meetings*. American Institute of Aeronautics and Astronautics, January 2007.

[3] Peter Huber and Patricia Seamount. X-31 high angle of attack control system performance. Technical report, NASA, 1995.

[4] M.V. Cook. *Flight Dynamics Principles*. Arnold, a member of the Hodder Headline Group, 338 Euston Road, London NW1 3BH, 1997.

[5] Iain K. Peddle. *Acceleration Based Manoeuvre Flight Control System for Unmanned Aerial Vehicles*. PhD thesis, Stellenbosch University, 2008.

[6] Upset Recovery Industry Team. *Airplane Upset Recovery Training Aid, Revision 2*, November 2008.

[7] Federal Aviation Administration. *Airplane Flying Handbook FAA-H-8083-3A*. U.S. Department of Transportation, Federal Aviation Administration, 2004.

[8] J.R. Chambers and S.B. Grafton. Aerodynamic characteristics of airplanes at high angles of attack. Technical Report NASA TM-74097, NASA, 1977.

[9] Robert M Seltzer and Glenn R Rhodeside. Fundamentals and methods of high angle-of-attack flying qualities research. Technical report, DTIC Document, 1988.

[10] C. Favre. Fly-by-wire for commercial aircraft: the airbus experience. *International Journal of Control*, 59(1):139–157, 1994.

[11] Y. Li, N. Sundararajan, and P. Saratchandran. Neuro-controller design for nonlinear fighter aircraft maneuver using fully tuned rbf networks. *Automatica*, 37(8):1293–1301, August 2001.

[12] T.S. Richardson, G.A. Charles, D.P. Stoten, M. di Bernardo, and M.H. Lowenberg. Continuation based control of aircraft dynamics. In *Decision and Control, 2003. Proceedings. 42nd IEEE Conference on*, volume 5, pages 4932–4938 Vol.5, 2003.

[13] Krzysztof Sibilski. Problems of manoeuvring at post-critical angles of attack - continuation and bifurcation methods approach. In *Aerospace Sciences Meetings*. American Institute of Aeronautics and Astronautics, January 2003.

[14] Aditya Paranjape and N Ananthkrishnan. Airplane level turn performance, including stability constraints, using ebac methodr. In *Guidance, Navigation, and Control and Co-located Conferences*. American Institute of Aeronautics and Astronautics, August 2005.

[15] Fen Wu, Ashok Gopalarathnam, and Sungwan Kim. Post-stall aerodynamic modeling and gain-scheduled control design. In *Guidance, Navigation, and Control and Co-located Conferences*. American Institute of Aeronautics and Astronautics, August 2005.

[16] C. D. Jones, M. H. Lowenberg, and T. S. Richardson. Tailored dynamic gain-scheduled control. *Journal of Guidance, Control, and Dynamics*, 29(6):1271–1281, November 2006.

[17] Nhan Nguyen, Kalmanje Krishnakumar, John Kaneshige, and Pascal Nespeca. Dynamics and adaptive control for stability recovery of damaged asymmetric aircraft. In *Guidance, Navigation, and Control and Co-located Conferences*. American Institute of Aeronautics and Astronautics, August 2006.

[18] Thomas Richardson, Mark Lowenberg, Mario DiBernardo, and Guy Charles. Design of a gain-scheduled flight control system using bifurcation analysis. *Journal of Guidance, Control, and Dynamics*, 29(2):444–453, March 2006.

[19] Jong-Yeob Shin and I. Gregory. Robust gain-scheduled fault tolerant control for a transport aircraft. In *Control Applications, 2007. CCA 2007. IEEE International Conference on*, pages 1209–1214, 2007.

[20] Jong-Yeob Shin and Christine Belcastro. Robustness analysis and reliable flight regime estimation of an integrated resilient control system for a transport aircraft. In *Guidance, Navigation, and Control and Co-located Conferences*. American Institute of Aeronautics and Astronautics, August 2008.

[21] Irene Gregory, Chengyu Cao, Enric Xargay, Naira Hovakimyan, and Xiaotian Zou. L1 adaptive control design for nasa airstar flight test vehicle. In *Guidance, Navigation, and Control and Co-located Conferences*. American Institute of Aeronautics and Astronautics, August 2009.

[22] Harry Kwatny, Robert Allen, and Gaurav Bajpai. Safe set protection and restoration for unimpaired and impaired aircraft. In *Guidance, Navigation, and Control and Co-located Conferences*. American Institute of Aeronautics and Astronautics, August 2012.

[23] S.R. Combs, K.G. Gousman, and G.J. Tauke. Pilot activated automatic recovery system on the f-117a. In *Aerospace Design Conference*. American Institute of Aeronautics and Astronautics, February 1992.

[24] Amitabh Saraf, Girish Deodhare, and Debasish Ghose. A feedback linearisation based nonlinear controller synthesis to recover an unstable aircraft from post-stall regime. In *Guidance, Navigation, and Control Conference and Exhibit*, pages 493–499. American Institute of Aeronautics and Astronautics, 1998.

[25] D. W. Sparks and D. D. Moerder. Optimal aircraft control upset recovery with and without component failures. In *Proceedings of the 2002 American Control Conference (IEEE Cat. No.CH37301)*, volume 5, pages 3644–3649 vol.5, May 2002.

[26] P. Kumar, P. Raghavendra, Tuhin Sahai, and Narayan Ananthkrishnan. Spin recovery of an aircraft using nonlinear dynamic inversion techniques. In *Aerospace Sciences Meetings*. American Institute of Aeronautics and Astronautics, January 2004.

[27] Dong-Chan Lee and M. G. Nagati. Momentum vector control for spin recovery. *Journal of Aircraft*, 41(6):1414–1423, November 2004.

[28] PK Raghavendra, Tuhin Sahai, P Ashwani Kumar, Manan Chauhan, and N Ananthkrishnan. Aircraft spin recovery, with and without thrust vectoring, using nonlinear dynamic inversion. *Journal of Aircraft*, 42(6):1492–1503, November 2005.

[29] Brian Dutoi, Nathan Richards, Neha Gandhi, David Ward, and John Leonard. Hybrid robust control and reinforcement learning for optimal upset recovery. In *Guidance, Navigation, and Control and Co-located Conferences*. American Institute of Aeronautics and Astronautics, August 2008.

[30] Nandan Kumar Sinha and D. M. K. K. Venkateswara Rao. Aircraft spin recovery using a sliding-mode controller. *Journal of Guidance, Control, and Dynamics*, 33(5):1675–1679, September 2010.

[31] Jean-Etienne Dongmo. Aircraft loss-of-control recovery using feedback linearization and high order sliding mode control. In *Guidance, Navigation, and Control and Co-located Conferences*. American Institute of Aeronautics and Astronautics, August 2012.

[32] Jean-Etienne Dongmo. Aircraft loss-of-control recovery using optimal high order sliding mode control with discontinuous high order observers. In *Guidance, Navigation, and Control and Co-located Conferences*. American Institute of Aeronautics and Astronautics, August 2012.

[33] Luis Crespo, Sean Kenny, Daniel Murri, and David Cox. Analysis of control strategies for aircraft flight upset recovery. In *Guidance, Navigation, and Control and Co-located Conferences*. American Institute of Aeronautics and Astronautics, August 2012.

[34] A. Ross, Geraldiine Edwards, Vladislav Klein, and James Batterson. Validation of aerodynamic parameters at high angles of attack for rae high incidence research models. In *Guidance, Navigation, and Control and Co-located Conferences*. American Institute of Aeronautics and Astronautics, August 1987.

[35] James G. Batterson and Vladislav Klein. Analysis of flight data from a high-incidence research model by system identification models. NASA Technical Paper 2940, NASA, November 1989.

[36] A. Jean Ross, Geraldine F. Edwards, Vladislav Klein, and James G. Batterson. Validation of aerodynamic parameters for high-incidence research models. *Journal of Aircraft*, 26(7):621–628, July 1989.

[37] Admire model, foi, http://www.foi.se/en/our-knowledge/aeronautics/ admire/ [online].

[38] Victoria Regenie, Donald Gatlin, Robert Kempel, and Neil Matheny. The f-18 high alpha research vehicle - a high-angle-of-attack testbed aircraft. In *National Heat Transfer Conference*. American Institute of Aeronautics and Astronautics, August 1992.

[39] T. Jordan, W. Langford, C. Belcastro, J. Foster, G. Shah, G. Howland, and R. Kidd. Development of a dynamically scaled generic transport model testbed for flight research experiments. In *AUVSI Unmanned Systems North America Symposium and Exhibition*, Anaheim, CA, 2004.

[40] John Foster, Kevin Cunningham, Charles Fremaux, Gautam Shah, Eric Stewart, Robert Rivers, James Wilborn, and William Gato. Dynamics modeling and simulation of large transport airplanes in upset conditions. In *Guidance, Navigation, and Control and Co-located Conferences*, page m. American Institute of Aeronautics and Astronautics, August 2005.

[41] Kevin Cunningham, John Foster, Austin Murch, and Eugene Morelli. Practical application of a subscale transport aircraft for flight research in control upset and failure conditions. In *Guidance, Navigation, and Control and Co-located Conferences*. American Institute of Aeronautics and Astronautics, August 2008.

[42] Patrick Murphy and Vladislav Klein. Transport aircraft system identification from wind tunnel data. In *Guidance, Navigation, and Control and Co-located Conferences*. American Institute of Aeronautics and Astronautics, August 2008.

[43] Patrick Murphy and Vladislav Klein. Transport aircraft system identification using roll and yaw oscillatory wind tunnel data. In *Guidance, Navigation, and Control and Co-located Conferences*. American Institute of Aeronautics and Astronautics, August 2010.

[44] B-C Chang, Harry Kwatny, Christine Belcastro, and Celeste Belcastro. Aircraft loss-of-control accident prevention: Switching control of the gtm aircraft with elevator jam failures. In *Guidance, Navigation, and Control and Co-located Conferences*. American Institute of Aeronautics and Astronautics, August 2008.

[45] Eric Groen, Wietse Ledegang, Joris Field, Hafid Smaili, Manfred Roza, Lars Fucke, Suzanne Nooij, Mikhail Goman, Michael Mayrhofer, Larisa Zaichik, Maxim Grigoryev, and Vladimir Biryukov. Supra - enhanced upset recovery simulation. In *Guidance, Navigation, and Control and Co-located Conferences*. American Institute of Aeronautics and Astronautics, August 2012.

[46] N. Abramov, M. Goman, A. Khrabrov, E. Kolesnikov, L. Fucke, B. Soemarwoto, and H. Smaili. Pushing ahead - supra airplane model for upset recovery. In *Guidance, Navigation, and Control and Co-located Conferences*. American Institute of Aeronautics and Astronautics, August 2012.

[47] Thomas Jordan, William Langford, Christine Belcastro, John Foster, Gautam Shah, Gregory Howland, and Reggie Kidd. Development of a Dynamically Scaled Generic Transport Model Testbed for Flight Research Experiments. In *AUVSI Unmanned Systems North America*, Arlington, 2004.

[48] Kevin Cunningham, John V Foster, Gautam H Shah, Eric C Stewart, Robin N Ventura, Robert A Rivers, James E Wilborn, and William Gato. Simulation Study of Flap Effects on a Commercial Transport Airplane in Upset Conditions. In *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, pages 1–14, San Francisco, California, 2005.

[49] Patrick C Murphy and Vladislav Klein. Transport Aircraft System Identification from Wind Tunnel Data. In *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, pages 1–14, 2008.

[50] Joseph R. Chambers. *Modelling Flight: The Role of Dynamically Scaled Free-Flight Models in Support of NASA's Aerospace Programs*. NASA SP 2009-575. National Aeronautics and Space Administration, 2009.

[51] Jacobus Engelbrecht, Simon Pauck, and Iain Peddle. Bifurcation analysis and simulation of stall and spin recovery for large transport aircraft. In *Guidance, Navigation, and Control and Co-located Conferences*. American Institute of Aeronautics and Astronautics, August 2012.

[52] Simon Pauck and Jacobus Engelbrecht. Bifurcation analysis of the generic transport model with a view to upset recovery. In *Guidance, Navigation, and Control and Co-located Conferences*. American Institute of Aeronautics and Astronautics, August 2012.

[53] Stephen Gill, Guilhem Puyou, Etienne Coetzee, M. Lowenberg, and B. Krauskopf. Bifurcation analysis of the nasa gtm with a view to upset recovery. In *Guidance, Navigation, and Control and Co-located Conferences*, pages –. American Institute of Aeronautics and Astronautics, August 2012.

[54] Steven H. Strogatz. *Nonlinear Dynamics and Chaos*. Westview Press, 2000.

[55] Aditya Paranjape, Nandan Sinha, and Narayan Ananthkrishnan. Use of bifurcation and continuation methods for aircraft trim and stability analysis - a state-of-the-art. In *Aerospace Sciences Meetings*. American Institute of Aeronautics and Astronautics, January 2007.

[56] James V. Carroll and Raman K. Mehra. Bifurcation analysis of nonlinear aircraft dynamics. *Journal of Guidance, Control, and Dynamics*, 5(5):529–536, September 1982.

[57] M.G. Goman, G.I. Zagainov, and A.V. Khramtsovsky. Application of bifurcation methods to nonlinear flight dynamics problems. *Progress in Aerospace Sciences*, 33:539–586, 1997.

[58] Peter A Cummings. Continuation methods for qualitative analysis of aircraft dynamics. Technical report, NASA, 2004.

[59] EJ Doedel, RC Paffenroth, AR Champneys, TF Fairgrieve, Yu, B Sandstede, and X Wang. Auto 2000: Continuation and bifurcation software for ordinary differential equations (with homcont). Technical report, California Institute of Technology, Pasadena, CA, USA, 2001.

[60] Jacobus Engelbrecht, Simon J. Pauck, and Iain K. Peddle. A multi-mode upset recovery flight control system for large transport aircraft. In *Guidance, Navigation, and Control and Co-located Conferences*, 2013.

[61] Jean-Jacques E. Slotine and Weiping Li. *Applied Nonlinear Control*. Prentice-Hall International Editions, 1991.

[62] Donald E. Kirk. *Optimal Control Theory*. Dover Publications, Inc., Mineola, New York, 2004.